

Master Thesis



Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Cybernetics

Manipulation Using a Compliant Robot

Bc. Jan Čermák

Supervisor: Ing. Vladimír Smutný, Ph.D.

Field of study: Cybernetics and Robotics

Subfield: Robotics

January 2020

I. Personal and study details

Student's name: **Čermák Jan**

Personal ID number: **420149**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Cybernetics**

Study program: **Cybernetics and Robotics**

Branch of study: **Robotics**

II. Master's thesis details

Master's thesis title in English:

Manipulation Using a Compliant Robot

Master's thesis title in Czech:

Manipulace za použití poddajného robotu

Guidelines:

1. Get familiar with the robot KUKA LBR controlled from Robot Operating System.
2. Find and study literature dealing with compliant motion control.
3. Implement assembly and disassembly of LEGO blocks.
4. Make experiments and conclusions.

Bibliography / sources:

- [1] Seung-kook Yun: Compliant manipulation for Peg-in-Hole: Is passive compliance a key to learn contact motion? June 2008 Proceedings - IEEE International Conference on Robotics and Automation DOI: 10.1109/ROBOT.2008.4543437
- [2] Daniel Leidner, Christoph Borst, ... Alin Albu-Schäffer: Classifying compliant manipulation tasks for automated planning in robotics, IEEE/RSJ International Conference on Intelligent... 2015 DOI:10.1109/iros.2015.7353607
- [3] Kalakrishnan M., Righetti L., Pastor P., and Schaal S.; Learning force control policies for compliant manipulation.; Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on IROS; 2011

Name and workplace of master's thesis supervisor:

Ing. Vladimír Smutný, Ph.D., Robotic Perception, CIIRC

Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **01.04.2019**

Deadline for master's thesis submission: _____

Assignment valid until: **19.02.2021**

Ing. Vladimír Smutný, Ph.D.
Supervisor's signature

doc. Ing. Tomáš Svoboda, Ph.D.
Head of department's signature

prof. Ing. Pavel Ripka, CSc.
Dean's signature

III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

Date of assignment receipt

Student's signature

Acknowledgements

I would like to thank my supervisor Ing. Vladimír Smutný, Ph.D. for so much useful advice and guidance, his patience and willingness to assist whenever I encountered trouble. My thanks also extend to Ing. Pavel Burget, Ph.D. for the opportunity to work with a real robot.

Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university thesis.

Prague, January 7, 2020

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 7. ledna 2020

Abstract

This thesis proposes techniques to assemble and disassemble LEGO bricks using a compliant robot. It is assumed that the brick's position is not precisely determined thus it is necessary to use force feedback. Some proposed techniques are based on the collision contact of the robot and its environment. The thesis also presents a design of robot fingers which ensures the firm grasp of a LEGO brick.

Keywords: LBR iiwa, Compliant manipulation, LEGO, Dis/Assemble, 3D print

Supervisor: Ing. Vladimír Smutný, Ph.D.

Abstrakt

Tato práce navrhuje postupy skládání a rozebírání LEGO kostek poddajným robotem. Předpokládá se, že pozice kostek není přesně známa a je nutné použití silové zpětné vazby. Některé navržené postupy jsou přímo založené na kontaktu robota a okolí. Tato práce prezentuje i návrh prstů manipulátoru pro pevný úchop LEGO kostky.

Klíčová slova: LBR iiwa, Poddajná manipulace, LEGO, De/Montáž, 3D tisk

Překlad názvu: Manipulace za použití poddajného robotu

Contents

1 Introduction	1
2 Motivation and related work	3
3 Working layout	5
3.1 Manipulator	5
3.1.1 ROS	6
3.1.2 Robot's software and compliant parameters	6
3.1.3 Gripper	7
3.1.4 Gripper fingers	7
3.2 Table layout	10
3.2.1 LEGO Base-plate	10
3.2.2 LEGO Brick	10
4 LEGO Problem	13
4.1 Terminology	13
4.2 Bricks versions	18
4.3 Placing configurations	19
4.3.1 Config 1 - Standalone stone .	20
4.3.2 Config 2 - Column	20
4.3.3 Config 3 - Pathway	21
4.3.4 Config 4 - Line	21
4.3.5 Config 5 - Step	22
4.3.6 Config 6 - Staircase	22
4.3.7 Config 7 - Neighbour	23
4.3.8 Config 8 - Corridor	23
4.3.9 Config 9 - Corner	24
4.3.10 Config 10 - Horseshoe	24
4.3.11 Config 11 - Box	25
4.4 Assembling	25
4.4.1 Trivial movement	27
4.4.2 Tilt movement	27
4.4.3 Spiral movement	29
4.4.4 Neighbour movement	31
4.4.5 Corner movement	33
4.4.6 Horseshoe motion	35
4.4.7 Box-spiral movement	36
4.4.8 Box-corner movement	38
4.5 Disassembling	39
4.5.1 Trivial movement	40
4.5.2 Tilt movement	41
4.5.3 Tilt-Repairing movement ...	42
5 Experiments	45
6 Conclusion	53
Bibliography	55

Figures

3.1 The working layout	5	4.26 Box-corner movement phases ..	38
3.2 Gripper Schunk EGP 50-N-N-B		4.27 The slip of the brick during the	
[14]	7	detachment of a stone	40
3.3 Fingers banding during gripping .	8	4.28 Trivial movement phases.....	41
3.4 Free space around the fingers in		4.29 Phases of the Tilt movement ..	41
attempt to grip the stone	8	4.30 Phases of the Tilt-Repairing	
3.5 Finger dimensions	9	movement	43
3.6 LEGO 630 bricks separator.....	10	4.31 The stretching force vector (\mathbf{F}) of	
3.7 Dimensions of the 2x4 LEGO brick		the <i>tensing</i> motion	44
[11].....	11		
4.1 A brick/stone parts terminology	14	5.1 Column configuration	46
4.2 Terminology of the brick		5.2 Line configuration	47
construction.....	15	5.3 Neighbour configuration	47
4.3 Versions of LEGO bricks.....	18	5.4 Corridor configuration	48
4.4 Summary of LEGO configurations	19	5.5 Corner configuration	48
4.5 Config. 1 - Standalone brick ...	20	5.6 Horseshoe configuration	49
4.6 Config. 2 - Column	21	5.7 Box configuration	49
4.7 Config. 3 - Pathway.....	21	5.8 Standalone configuration	50
4.8 Config. 4 - Line	22	5.9 Staircase configuration	51
4.9 Config. 5 - Step	22	5.10 Neighbour configuration	52
4.10 Config. 6 - Staircase	23		
4.11 Config. 7 - Neighbour	23		
4.12 Config. 8 - Corridor.....	24		
4.13 Config. 9 - Corner	24		
4.14 Config. 10 - Horseshoe	25		
4.15 Config. 11 - Box	25		
4.16 Trivial movement phases.....	27		
4.17 Consequences of imprecise Z			
height	27		
4.18 Tilt movement phases	28		
4.19 Angle of the second motion and			
marked hit contact of the fifth			
motion	28		
4.20 The correctly and wrongly			
oriented gripper when attaching the			
stone into the goal position which is			
situated in close proximity to			
adjacent brick.	29		
4.21 Spiral movement phases	30		
4.22 Neighbour movement phases ..	32		
4.23 Corner movement phases	33		
4.24 Horseshoe movement phases...	35		
4.25 Box-spiral movement phases. In			
images, there is a transparent skewed			
brick which enables a position of the			
stone being visible in each motion.	37		

Tables



Chapter 1

Introduction

In recent times, robot manufacturers have come up with collaborative robots. The current trend in industry is the using of robots together with human operators to collaborate on a task. During the task, both human and robot are in close proximity and possibility of mutual contact is highly probable. For this reason, robots are equipped with mechanisms to sense external forces. Thus, the robot can recognize a potential collision within its environment and react to this case e.g. acquiesce on external forces and move in the direction of the force. If a robot is equipped with a mechanism for an external force sensing which has force feedback used to affect the robot's mechanical state, then this robot is being referred to compliant.

This diploma thesis is focused on robotic manipulation using a compliant robot. We will discuss the setting of stiffness, damping ratio and force conditions. For this research, we have used collaborative robot KUKA LBR IIWA 14 with IO-electric flange configuration. The assignment of this thesis is to handle the assembling and disassembling of LEGO bricks. The motivation of using this robot and LEGO bricks for demonstrating this issue is that the assembling of LEGO requires a high accuracy of brick placing, whereas the robot's repeatability is worse.

A LEGO base-plate is manufactured with very high precision. On the other hand, the Cartesian space of an uncalibrated robot is highly deformed. Let's describe this topic closer.

The classic LEGO base-plate we use has 32x32 studs [13], has regular perpendicular structure of studs in the Cartesian space. In other words, the adjacent lines of studs are highly parallel and are less skewed and curved. On the other hand, the robot's operational space is much less precise in the sense of perpendicularity and line parallelity. When a robot is trying to perform a linear movement, the real movement within the world's (\equiv absolute) Cartesian space is partially curved in all directions. The deviation can easily achieve a value of 0.5 mm. When a robot tries to perform two adjacent linear movements, in distance of two LEGO studs, the tendency of line distortion is the same, e.g. the ends of the lines can be more curved than the middles, only if the same joint configuration is held. If we use different robot's configuration, where the robot meant to perform the same trajectory but where the joints have different values, the distortion of each line has a different characteristic.

This imprecision in the Cartesian space is caused by inaccurate robot's parts manufacturing and assembling. In general, these unknown differences are unique for each robot. They are often very small, but total impact of these differences in kinematic equations result in high inaccuracy of a robot's tip.

To eliminate this inaccuracy, it is possible to perform a kinematic calibration, which determines actual values of these parameters. The standard tool used for the kinematic calibration is a laser tracker. Kinematic calibration is not intentionally used in this work to have a robot with adequate inaccuracy. We are trying to find techniques that would solve the task independently of described distortions and inaccuracies.

In fact, people are less accurate than any industrial robot and yet assembling LEGO is quite easy for them. The reason is that we do not focus too much on precise positioning but we focus on tactile feedback from finger tips. People are also not able to determine the right brick assembling position just from only one haptic contact. It is meant that just from one contact of the manipulated brick with another brick they are not able to determine the right correction of manipulated brick position. People rather use a form of compliance during this task.

Our solution is also inspired by human manipulation. We do not use tactile sensors or robotic fingers characteristic for human manipulation. We use a robot's in-built force feedback which substitutes for the tactile sensors.

The alternative approach to this problem is using a very accurate robot, which would be kinematically calibrated. This approach is an industrial one. The preferred cheaper solution corresponds to using of less precise robots where robot's imprecision of its positioning is solved programmatically.

Chapter 2

Motivation and related work

This thesis was partially inspired by the work of Jérémy Brouillard, a B.Eng. student at the ÉTS university in Canada [1]. They used a precise delta robot without force feedback to assemble and disassemble LEGO bricks. They are capable of fine brick placement into positions where the solid brick underneath is missing. The theoretical work of Bruyninckx, Dutré and Schutter [2] determined the analytical solution for the cylindrical peg and hole problem. They determined that the problem has three DOFs and each DOF they assigned to the robot's joint or joints. The work [3] also identified and categorized the peg and hole problem of a cylindrical object. In this work, an 8-DOFs manipulator was used, without force feedback. The alignment of the cylindrical object and the hole is measured by Kinect. Another work of the same authors [4], also concerned with the peg and hole problem, only uses a compliant robot which does not use any force sensor nor vision system in order to measure the precise hole position. The work aims to use position feedback to recognize any improvement in the robot's position and orientation. The peg has a block shape. The similar problem is also the main part of the work [5]. It determines the equations to compute various frictions when a screw is inserted into the hole. They used a robot which had an added force sensor placed on its tip. This construction eliminates the variance of the measured force on the robot's configuration and position. This more complex work [6] is focused on efficiency and the speed of the peg and hole problem. It combines force feedback and visual feedback to autonomously solve this issue. The paper [7] designs the compliant manipulation of multiple robots. It proposes controllers for an object position and forcing of robots on the object. The paper [8] presents a designed force sensor useful for robot grippers. The sensor can be placed on the fingers and thus the collisions or haptic feedback can be gained exactly from the contact position of the manipulator and the manipulated object. The sensor is hemispherical shape and it can measure a force on three axes.

The demand of an object placing into an unreachable position due to the obstacles solves the work [9]. This work presents the object positioning without gripping it. The object motion is mediated by pushing it and using gravity force.

Most of the recent works are focused on visual feedback in order to obtain



the right position. This work uses only force feedback to influence the object motion.

Chapter 3

Working layout

This chapter describes the used equipment in the figure (3.1). The original component is the gripper finger whose construction is described in the section (3.1.4).

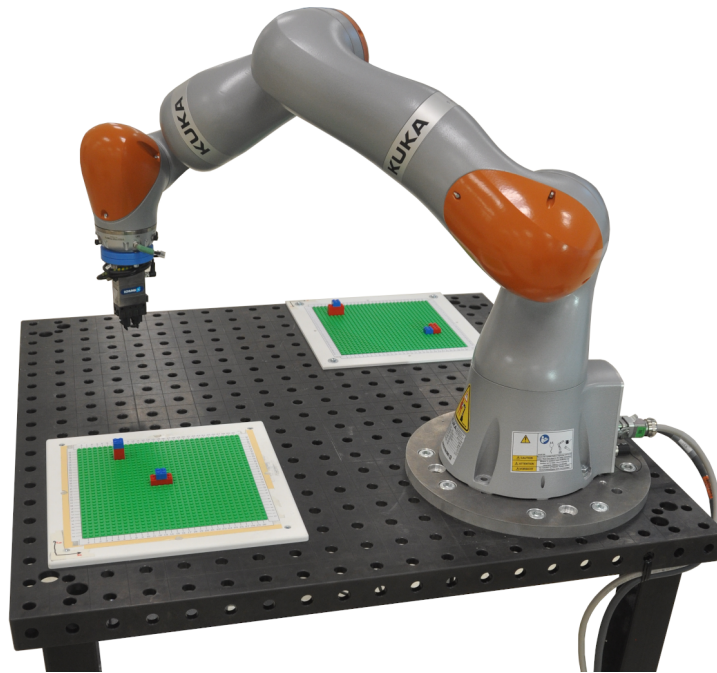


Figure 3.1: The working layout

3.1 Manipulator

As previously mentioned in the introduction chapter, we have used the collaborative robot KUKA LBR IIWA 14 with IO-electric flange. The robot is 820 mm in length and it can manipulate a payload with the maximal weight of 14 kg. This robot differs from other collaborative robots as it can apply the technique of external force measuring. Each of the robot's joints is equipped with a strain gauge which is capable of measuring the joint torque

more precisely than e.g. motor current feedback. For that reason, the robot actively uses the measured external force for motion influencing instead of just as a safety stop feature.

Generally, the external force measuring has too high variance for LEGO manipulation. But for short distances with identical joint configurations, the deviation of the external force measuring is very small and thus usable for LEGO manipulation. The standard deviation of the measured external force also increases with temperature and the time, due to the robot heating up.

■ 3.1.1 ROS

ROS is an abbreviation of the Robot Operating System that is a middleware which provides a communication tool for hardware and software. Moreover, it provides a wide palette of tools for robot control, simulation environments, planning algorithms and so on. It is frequently used in an experimental robotic. For the robot, its manufacturer provides a ROS package named FRI. This package allows controlling and commanding the status of a robot from an external computer.

The using of ROS and FRI was the original intention. Problems with the FRI availability and robot control resolved into ROS exclusion from this task.

■ 3.1.2 Robot's software and compliant parameters

The robot is programmed in KUKA SunriseWorkbench-1.16 based on Eclipse IDE and KUKA WorkVisual-5.0. WorkVisual serves for gripper control signals mapping, while the Workbench is the SDK in Java.

In the Workbench, there are implemented programs and application data. The application data covers taught robot positions, bases, gripper coordinate systems and so on.

The motion of the robot is controlled according to the selected coordinate system. These coordinate systems are often situated in the robot's fingers.

The robot can set a compliant mode for its motions or more precisely for the controlled coordinate system. The main parameters of the compliant mode are stiffness, damping ratio and bias. The motion in each axis of the coordinate system can be influenced. The stiffness and damping ratio are modeled by a virtual spring. The spring stiffness determines the extent to which the robot yields to an external force and deviates from its planned path. The spring damping determines the extent to which the virtual spring oscillate after deflection [15]. In other words, the stiffness is a property of the virtual spring expressing its strength while the damping ratio is a property of a system expressing the ratio how the system impedes to return the deflected spring back. The stiffness value range for the translation axes is from 0 N/m to 5000 N/m and for the rotation axes from 0 Nm/rad to 300 Nm/rad. The damping ratio value range is from 0.1 to 1.0, which is the critical damping. The bias represents an additional force in a selected direction, e.g. pressure force in Cartesian Z direction.

■ 3.1.3 Gripper

Many producers exist and, additionally, many more variants of grippers with various adjustments. The most known producers are Schunk, Zimmer, Festo, Robotiq, Röhm, Gripper Systems. Most of the grippers used are either electric or pneumatic powered. In fact, we have a pure electric flange mounted on the tip of the robot so we have to use an electric gripper. We have used a Schunk EGP 50-N-N-B with adjustable grip force within a range of 54 N to 215 N according to the datasheet [14].

The gripper is controlled by two logical signals, thus it has 4 states. The states are open, close, loosen and forbidden. The loosen state is used to open the gripper's jaws to the required distance. The opening time was measured as 165 ms from the close state. If we want to open the gripper to grip the stone, we have to stop the opening process of the gripper from the close state by setting the loosen state when the time 25 ms elapses. This time was measured experimentally. This is the way we set the gripper's jaws distances.

The gripper has one motor, so the jaws are positioned simultaneously. The gripper is attached to the flange of the robot.



Figure 3.2: Gripper Schunk EGP 50-N-N-B [14]

In our experiments, the gripping force is set to the maximal value, which is not necessary with the designed fingers described in the section (3.1.4). A reasonable setting is 50% of the maximal gripping force.

■ 3.1.4 Gripper fingers

The fingers were designed for 2x2 LEGO bricks. The current fingers are the third version inspired by previous versions. The previous two versions were designed by Jiří Medonos. The current fingers were printed on a 3D printer, which prints in layers, from an ABS string. The direction is very important for finger printing. In the figure (3.5), we can see image (c), which shows the Left view of the finger, which is the bottom side during the 3D printing. The printing direction of the layer is from up to down. This is because the

finger tip is tiny and skewed 5.5° from a gripper ceter axis. Other orientations would not lead to the correctly printed finger.

The finger dimensions are set for the used gripper so when the 2x2 brick is gripped and the gripper is completely closed, the finger tips are bent to maximize gripping area and they apply force to the brick in an effort to straighten back. See the figure (3.3).

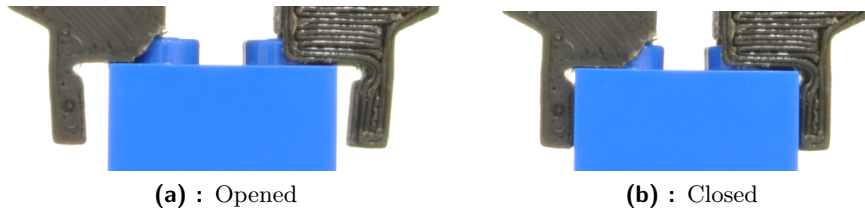


Figure 3.3: Fingers banding during gripping

The figure (3.4) shows the empty space around the fingers in an attempt to grip the blue stone. It is opened in order to compensate for small deviations of the stone's position.

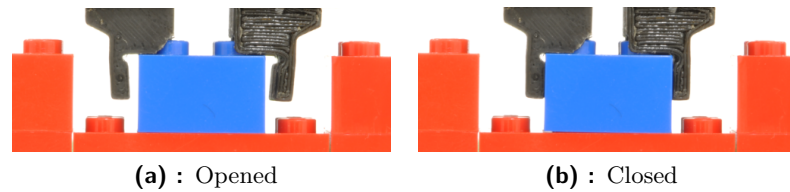


Figure 3.4: Free space around the fingers in attempt to grip the stone

The fingers are mounted to the gripper.

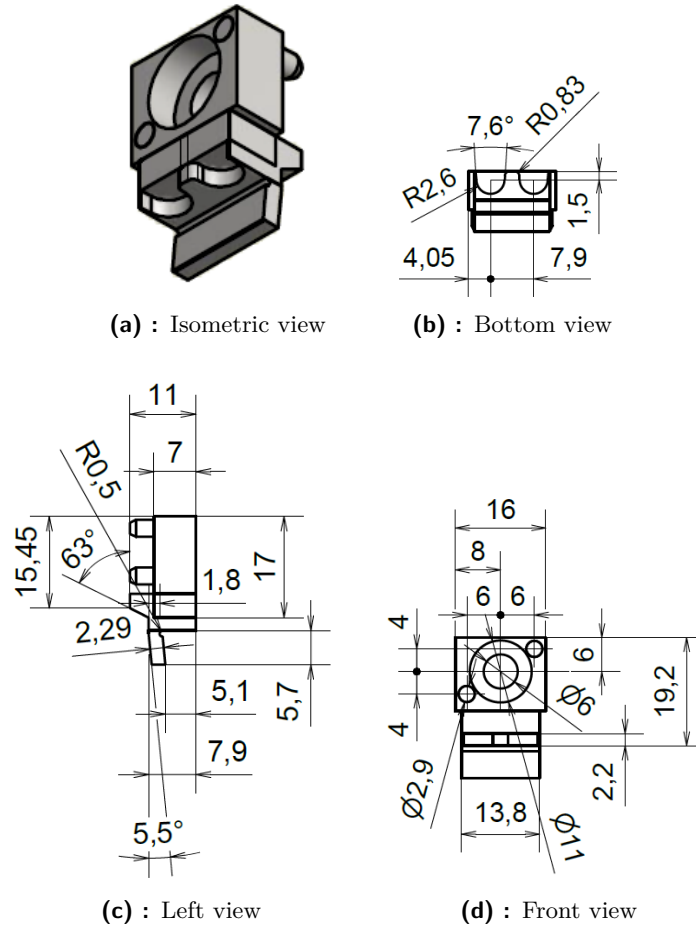


Figure 3.5: Finger dimensions

The predecessor of these fingers did not have skewed finger tips. There was a problem with the brick slipping during manipulation. Neither attaching rubber nor roughening the surface with a file fixed this issue. That is the reason for having the skewed finger tips in this version.

Further in the text, there is a chapter with LEGO configurations. It is mentioned that we cannot disassemble the Corner (4.3.9) and the Horseshoe (4.3.10) LEGO configurations due to the design of these fingers. To detach these configurations, a new design of fingers is necessary. It is possible to be inspired by the official LEGO separator, as shown in the figure (3.6). The separator uses a leverage effect for the brick's detaching, where the brick's wall and the stud-walls are fixed by the tool.



Figure 3.6: LEGO 630 bricks separator.

3.2 Table layout

The table serves as the substructure for the operational space. The robot and LEGO base-plates are mounted to it. We have used the welding table Siegmund 16 Basic because of its high stability and stiffness. The great advantages of this table are the prepared mounting holes and the etched grooves. The grooves are very useful for both the determination of the LEGO base-plate's base and the calibration of the robot's orientation.

3.2.1 LEGO Base-plate

The LEGO base-plate, which we have used, has the size of 32x32 studs [13]. It is a regular perpendicular structure of studs in the Cartesian space. The base-plates are mounted to the 3D printed white substructure which is mounted to the table as shown in the figure (3.1).

Base-plates are used as brickwork (explained in the terminology part) to which the bricks are positioned. Brick positions are computed from corners of this base-plate.

3.2.2 LEGO Brick

The LEGO brick dimensions are depicted in the figure (3.7).

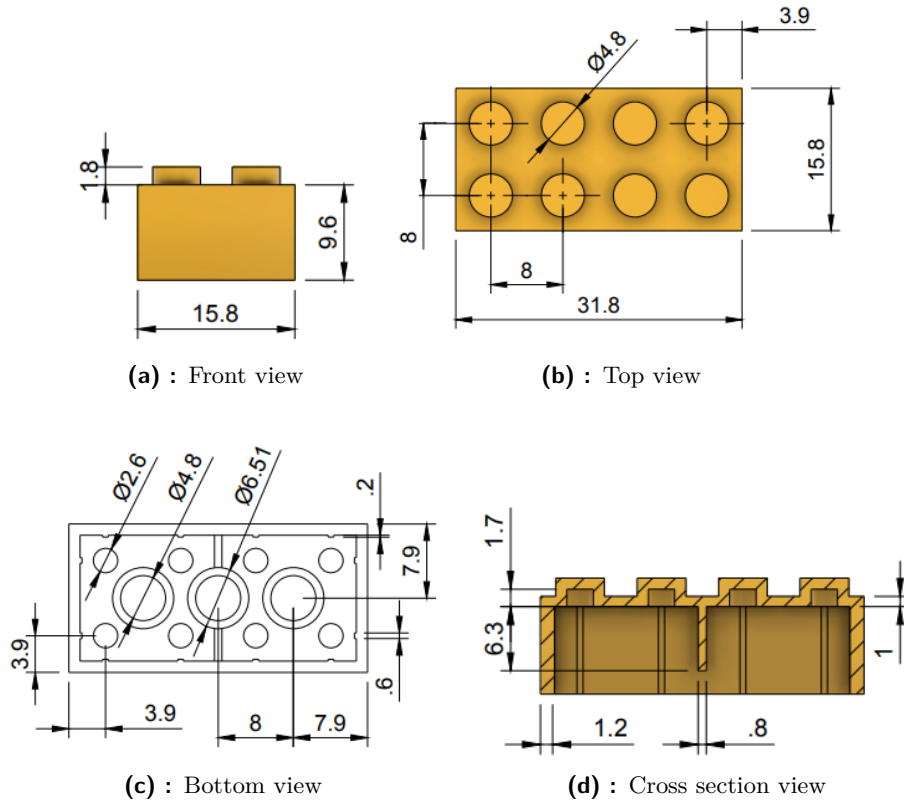


Figure 3.7: Dimensions of the 2x4 LEGO brick [11].

In image (3.7) we see the dimensions of the LEGO brick version 1, which is explained further in section (4.2). Also, the used terminology for the brick's parts is mentioned in the same chapter.

Chapter 4

LEGO Problem

This chapter describes the main issues and their solutions. It defines terminology of bricks parts and describes the used motions of proposed movements. The first section (4.1) defines the terminology and motions. The second section (4.2) mentions different LEGO block designs and their usage problems. The third section (4.3) divides LEGO layout to simple configurations and describes them. The fourth section (4.4) introduces proposed assembling movements and their ideas. And the fifth and final section (4.5) introduces the proposed disassembling movements.

This thesis deals with 2x2 LEGO bricks assembly and disassembly and all mentioned values are set to handle this brick size. It is mentioned in section (4.2) that the used bricks are of version 1.

In the chapter experiments (5), there is a small part describing the effects of some other brick sizes.

4.1 Terminology

For better understanding and easier naming of the following motions, it is necessary to define a terminology. This terminology is valid only for this thesis because on the internet there was not found any general naming convention of LEGO brick parts. Let us call the LEGO brick parts according to figure (4.1). Image (*a*) describes the main division of a LEGO brick while the second image (*b*) describes the brick's parts in detail. The image (*b*) defines names of edges and flat areas that are frequently used in following motion descriptions.

To clearly distinguish the brick which is handled by the gripper in the movement context, let us refer to this brick as **the stone**.

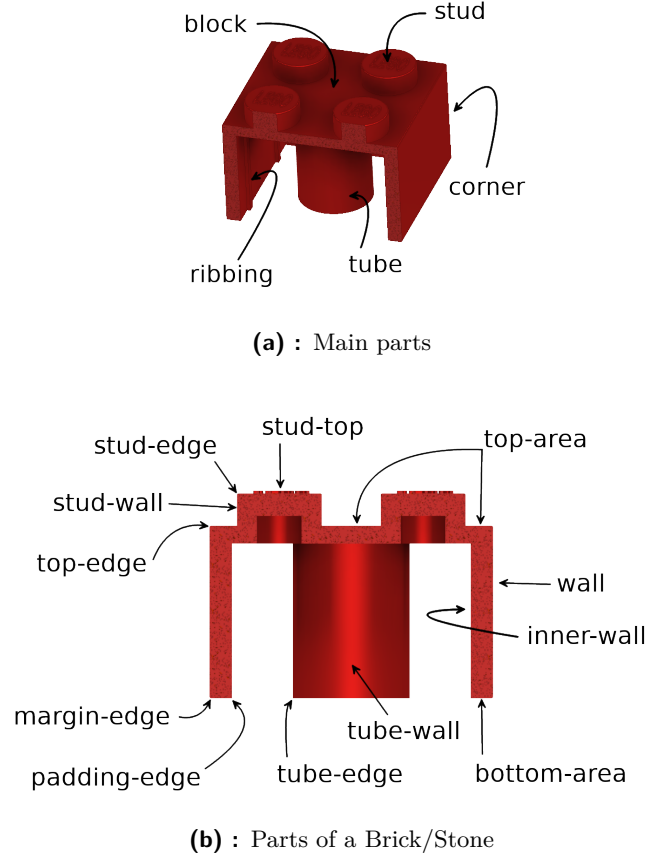


Figure 4.1: A brick/stone parts terminology

The figure (4.2) is changed from the model [12]. The figure (4.2) also consists of two images where the first image *a* defines brick layers and the second image *b* defines relations between the attached stone and other bricks.

Let's explain image *a* in the figure (4.2). The goal layer is the layer where the stone has its goal position according to the assembling procedure or start position according to the disassembling procedure. The layer starts at the attached stone's bottom-area and ends at the stone's top-area. The lower layers are those which are placed below the attached stone's bottom-area. The upper layers are those which are placed above the attached stone's top-area.

Image *b* in the figure (4.2) divides the depicted bricks into three categories. The stone is always the only one which is moving. The attached stone is the stone that is assembled in the goal position. **Adjacent bricks** are those bricks which are already attached in the goal layer and between them and the real goal position there are no other bricks present. Bricks that aren't either adjacent bricks or the stone are **brickwork**.

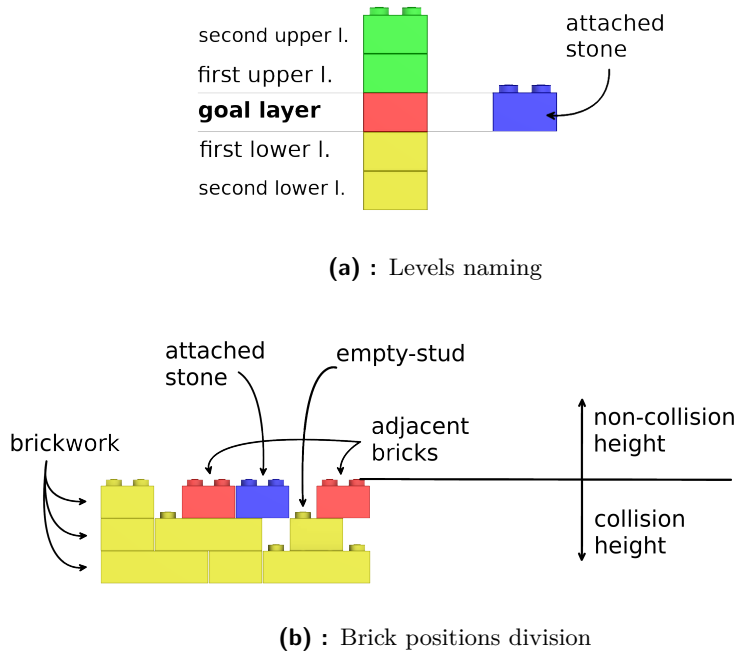


Figure 4.2: Terminology of the brick construction

Let us define used phrases:

- **Expected goal position.** It is an expected goal position of the stone where it should be attached.
- **Real goal position.** It is a real goal position of the stone where it is aimed to attached.
- **Grip position.** It is a position where the stone is grasped by the fingers.
- Let's distinguish the coordinate systems of the gripper and the base-plate. The axes of base-plate are labeled uppercase (X, Y, Z) while the axes of the gripper are lowercase (x, y, z) . The axes X, Y are axes of the base-plate area while the Z is the perpendicular axis to the base-plate area heading upwards.
- Motion directions have been shortcut by axis and the direction. **Z -downwards** leads from upper Z height towards the base-plate while **Z -upwards** is vice versa.
- **Non-collision height.** It is a height in Z where any part of the robot or the stone cannot come into contact with other bricks.
- **Collision height.** It is a height in Z where the fingers of the gripper or the stone can come into contact with brickwork or adjacent bricks.
- When a motion is **force conditioned**, it means that the motion is interrupted when a force-limit in a specific direction is exceeded.

- When a motion is **compliant** or it is in **compliant mode**, it means that the motion has set at least one compliant parameter which modifies the motion. See (3.1.2).
- **Empty-stud.** It is a stud that is not integrated into another brick. It is possible to attach a brick onto this stud. In the context of the stone, the empty-stud row is the closest row of empty studs nearby the stone.
- **Solid brickwork.** This means that the goal position of the stone has such solid brickwork that a large Z-downwards force does not disassemble the brickwork.
- When the stone and the adjacent brick are so close that there is not any empty-stud between them, they are in **close proximity**.
- When an action is labelled as **sensitive**, it means that the action is not robust in a specific way and the relatively small deviation of the normal status can cause an action failure.
- To **align stone** means that the stone is rotated to align orientation of x,y,z and X,Y,Z coordinate systems.
- ϕ is the experimentally obtained maximal deviation of the real positions on the base-plate. The value is $\phi = 0.5$ mm in all directions.
- **Motion.** Each proposed movement consists of motions that describe the single-purposed robot's motion.

Motions naming used in the sections of proposed movements:

- **Approaching.** It places the stone above the expected goal position. The stone is situated at a non-collision height during the whole motion. The height between the expected goal and the stone is equal to $h_{nc} = h_s + 3 * \phi = 12.8$ mm, where h_s is the height of the stone. In the case of reasonably higher imprecision in the height, the value $2 * \phi$ is intended to cover this deviation. This motion is the first executed motion in all proposed movements.
- **Landing.** It places the stone above the real goal position and partially attaches it onto the studs of the brickwork.
- **Leaving.** It places the gripper Z-upwards into non-collision height equal to the *approaching* height. This motion is the last executed motion in all proposed movements.
- **Attaching.** It places the stone into the real goal position. The stone is fixed in this position.
- **Tilting.** It tilts the stone around the x or y axis of the stone's padding-edge.

- **Leveling.** It determines the real Z position. The stone is moved Z-downwards until it comes into contact with the brickwork which means that this motion is force conditioned in the Z direction.
- **Gliding.** Shifts the stone in the defined direction along the XY plane until it comes into contact with the brickwork studs. This motion is force conditioned in the direction of the motion which adapts to the real X or Y position and moreover it is compliant in the XY plane in the perpendicular direction to the motion to adapt its position in the real Y or X position according to the stone's orientation.
- **Spiral.** It executes the inbuilt spiral function within a compliant mode. This motion aims to overcome a deviation in the XY plane in both directions.
- **Loosening.** It is a pure compliant motion without a directed motion, which means that the target position is equal to the start position. However, this position is extremely unstable under the small external force.
- **Nearing.** It places the stone into a collision height.
- **Touching.** It moves the stone in the XY plane until the force-limit in that direction is exceeded. The motion direction is labeled in the name e.g. touching-X-pos means that the collision occurs along X-positive direction.
- **Reinitiating.** It places the stone above the expected goal position as *approaching* motion, but it starts in a collision height.
- **Returning.** It moves the brick into the start position of a previous motion.
- **Arising.** It is a Z-upwards motion of the empty gripper in the non-collision height.
- **Arranging.** This motion is general and it sets the gripper position and orientation suitable for a next motion.
- **Lifting.** The gripper gripping the stone is moved Z-upwards.
- **Dropping.** The stone is dropped from a height.
- **Slanting.** Similar to tilting, but the stone is tilted around its corner in a defined axes.
- **Adapting.** The robot aligns the stone in order to fit it into the bounded position. The performed motion is compliant.
- **Gripping.** The stone is grasped by the fingers.

- **Tensing.** This is a very short motion or force acting motion whose purpose is to break the attachment between the stone and the underlying brickwork.
- **Straightening.** It aligns the stone's position purposely over its part.
- **Repairing.** This motion is similar to the *leveling* motion but the purpose is to resolve the issue of partially detached brickwork.

4.2 Bricks versions

This section describes revealed LEGO bricks versions and their differences. We have found these three versions which are depicted in the figure (4.3). This figure shows 6 images of brick bottoms. The upper images ((a),(b),(c)) show unchanged bottoms of each version while the lower images ((d),(e),(f)) show the same images but with highlighted differences. The brick versions are aligned in columns.

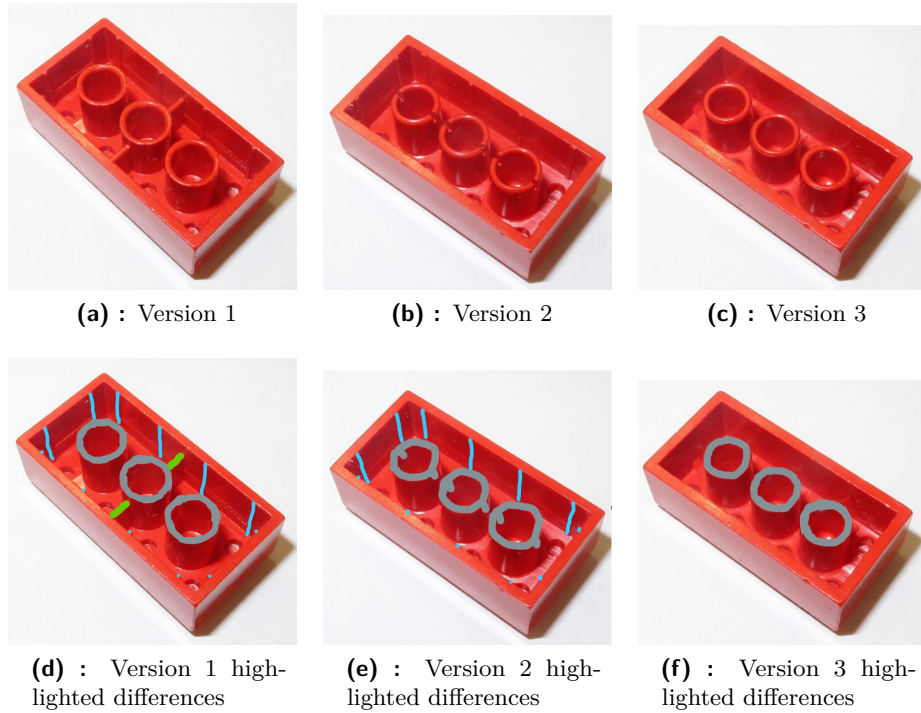


Figure 4.3: Versions of LEGO bricks

The front sides of all three versions are equal. The major differences are in the presence of side ribs, reinforcement ribs and in the beveling of tube-edges.

Version one has ribbing on each inner wall where the stud-walls of embedded bricks come into contact. These ribs eliminate free space and cause resistance during assembling and disassembling procedures. In the middles of longer walls, reinforcement ribs are present, which suppress the curvature of the

longer walls by increasing stiffness. Version one has minimally sloped tube-edges.

Version two has also ribs on its inner walls, but they are less thick. It has no reinforcement ribs and the tube-wall has two notches. The tube-edges are more inclined than those in version one. The third version only has highly inclined tube-edges.

Each version of the LEGO brick requires a different push force to be attached or detached. This force differences are perceptible by hand.

Most of the used bricks are of the version one type. This brick version also requires the largest force to be manipulated.

For example, the measured forces of non-collision attachment are approximately 6 N, 4 N and 1 N for versions 1, 2 and 3. These values are a rough estimation because they were measured by the robot. The standard deviation of the robot's force inaccuracy is 2 N.

4.3 Placing configurations

This section describes LEGO bricks configurations. All configurations are shown in the figure (4.4), where the red bricks restrict the blue stone. Each configuration can occur in four orientations in relation to the robot's position. Each orientation is rotated 90° from the previous one. Some orientations are equal for certain configurations.

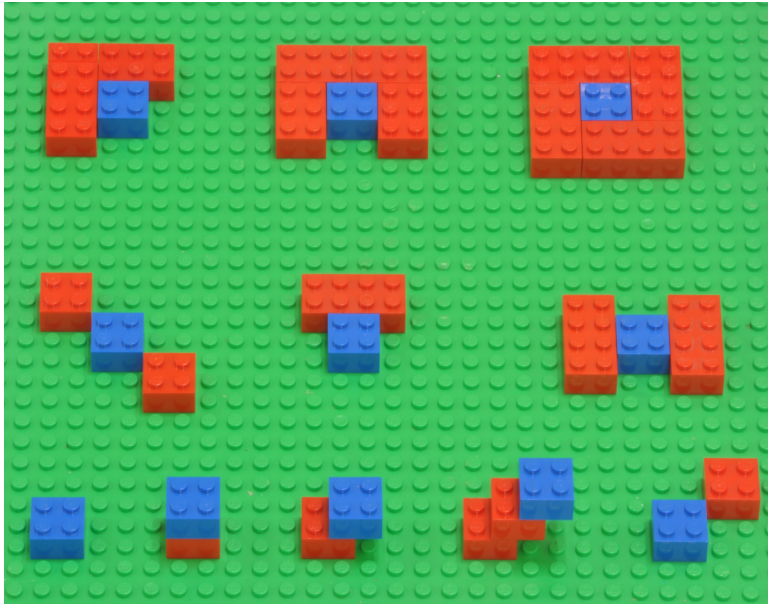


Figure 4.4: Summary of LEGO configurations

We expect that stones are assembled layer by layer. It means, the assembling procedure can be blocked by adjacent bricks in the goal layer but never bricks in the upper layer.

The following sections, which describe the LEGO configurations, always have a figure which consists of three images. The first image *(a)* shows the obvious example of the configuration, while the third image *(c)* shows a more difficult example. The second image *(b)* shows the brick layout of the configuration's goal layer. The used colours represent a specific kind of bricks. The blue colour marks the stone, the white colour marks empty-studs, the red colour marks the adjacent bricks and the yellow colour marks the position which is insignificant.

4.3.1 Config 1 - Standalone stone

The first configuration is a standalone stone as shown in the figure (4.5). The stone is surrounded by at least one row of empty-studs around while the brickwork is solid.

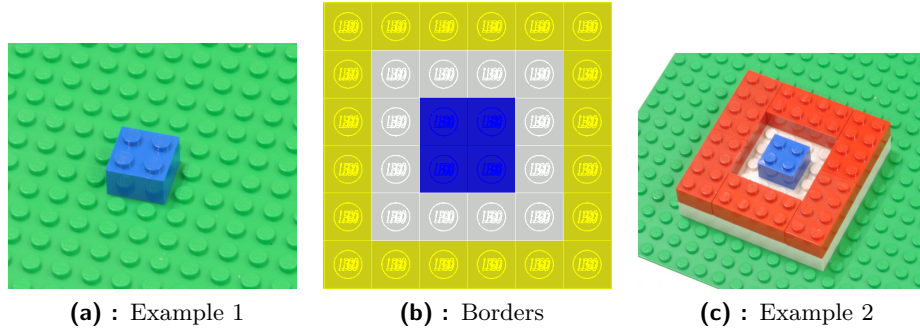


Figure 4.5: Config. 1 - Standalone brick

In the third image (c), the stone is surrounded by adjacent bricks which are not directly upon the base-plate.

This configuration is equal for all four orientations so we do not distinguish orientations.

4.3.2 Config 2 - Column

This configuration creates a column. The stone is attached to a brick or brickwork that has an equal number of studs as the stone while the brickwork is solid. The stone and the underlying brick have at least one row of empty-studs around.

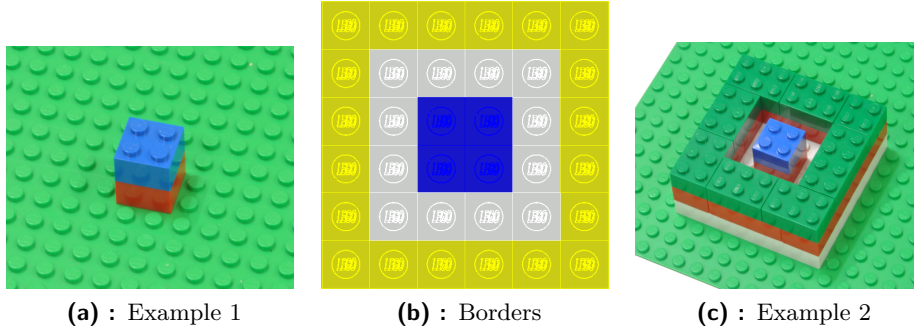


Figure 4.6: Config. 2 - Column

We consider this configuration equal to the configuration (4.3.1) because borders in the images (b) are very similar and from the view of the assembling procedure these two configurations are not distinguished as well as their orientations.

Image (c) shows the closest distance to adjacent bricks.

4.3.3 Config 3 - Pathway

The stone has at least one row of empty-studs around, except for at least one of the stone's corner which is in close proximity to the adjacent brick as shown in image (a) of the figure (4.7). For better understanding, see image (b). Image (c) shows the minimum number of empty-studs around the stone in order to be able to refer to this LEGO configuration as the Pathway.

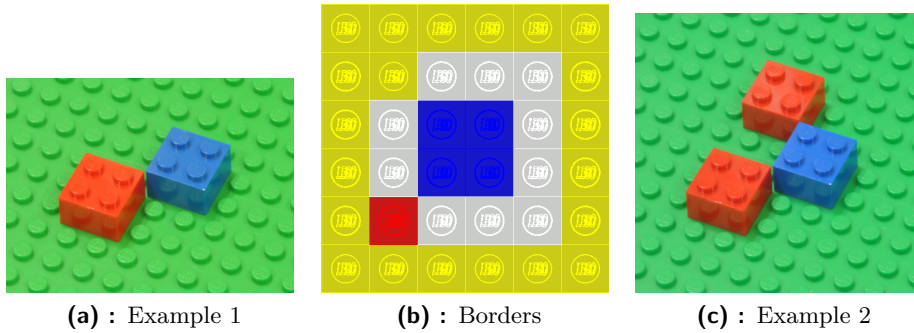


Figure 4.7: Config. 3 - Pathway

In fact, this configuration restricts the gripper motion in one DOF in the case of imprecise positioning.

4.3.4 Config 4 - Line

The stone has at least one row of empty-studs around it, except for its two diagonal corners which are in close proximity with adjacent bricks as image (a) of the figure (4.8) shows. Image (c) shows the case with the minimum empty-studs around the stone.

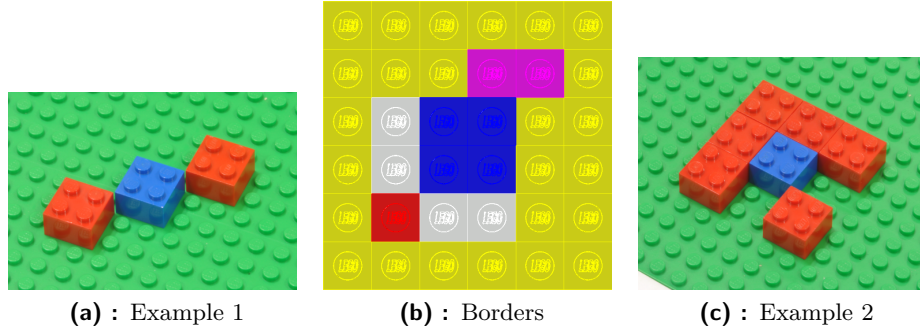


Figure 4.8: Config. 4 - Line

This configuration restricts the gripper motion in all DOFs in the case of imprecise positioning. In image (b), there are purple coloured positions which means that the adjacent brick is placed onto at least one position of these.

4.3.5 Config 5 - Step

The *Step* configuration occurs when one bottom-area of the attached stone is not in contact with the brickwork. The attached stone has one row of empty-studs around within the goal layer and one row of empty-studs under the bottom-area and its studs in the first lower layer as shown in the figure (4.9(a)). The brickwork is solid.

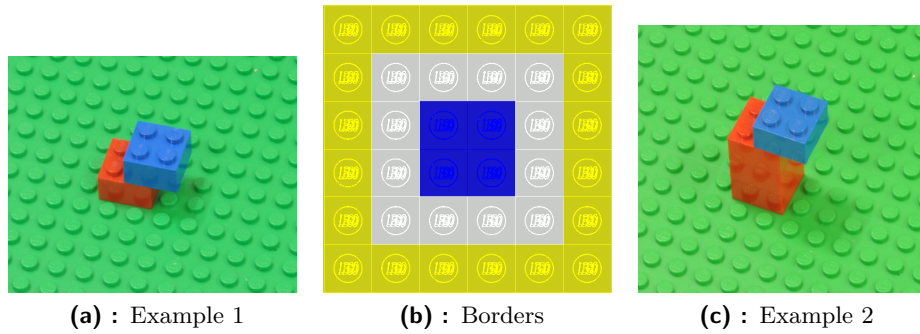


Figure 4.9: Config. 5 - Step

Image (c) shows the Step configuration in a higher layer.

4.3.6 Config 6 - Staircase

The *Staircase* configuration is a stricter version of the configuration (4.3.5). The brickwork is not solid and thus under a high Z-downwards force it collapses. The stone has at least one row of empty-studs around as shown in the figure (4.10).

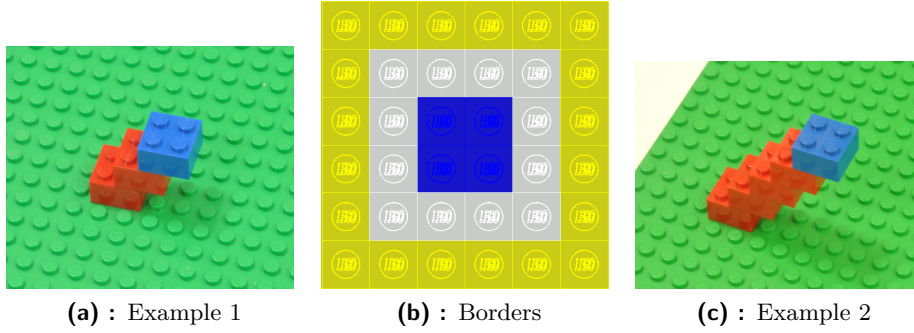


Figure 4.10: Config. 6 - Staircase

Image (c) shows an extreme case, which is very difficult to assemble.

4.3.7 Config 7 - Neighbour

This Neighbour configuration occurs when one wall of the stone and one wall of the adjacent brick are in close proximity. The other three walls of the stone have at least one row of empty-studs around. The figure (4.11) depicts this configuration.

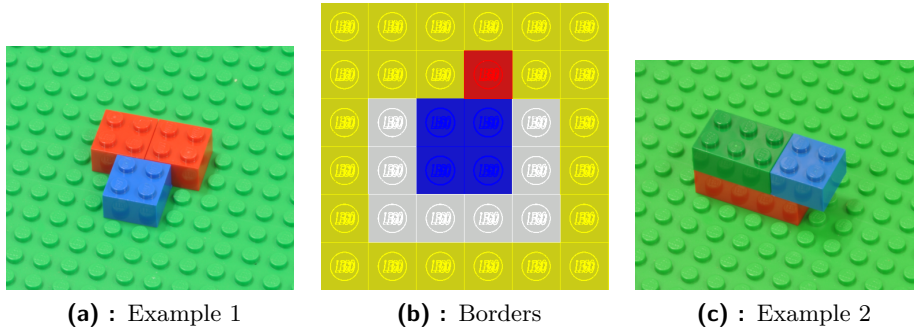


Figure 4.11: Config. 7 - Neighbour

In image (c), there is a case which looks like Step (4.3.5) LEGO configuration but it is not due to the adjacent brick in being in close proximity.

4.3.8 Config 8 - Corridor

This is the Corridor configuration. Any two parallel walls of the stone have at least one row of empty-studs while the other parallel walls are in close proximity to the adjacent bricks. The configuration is shown in the figure (4.12).

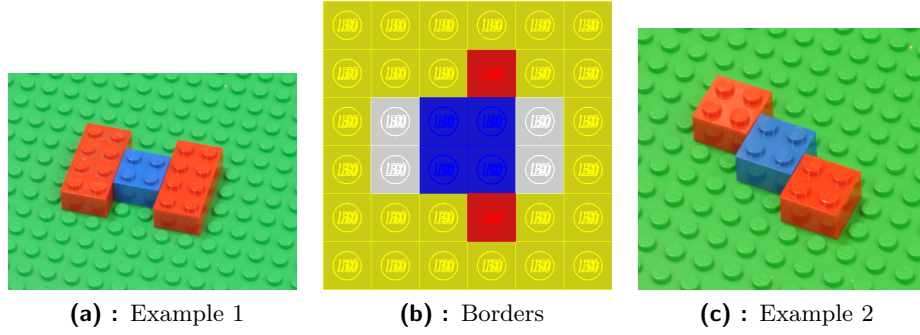


Figure 4.12: Config. 8 - Corridor

Image (c) shows the minimal contact area of this configuration. This configuration has only two orientations.

4.3.9 Config 9 - Corner

This is the Corner configuration. The adjacent bricks create a corner shape. We are interested in the inner corner where the stone is placed. The two walls of the stone which share a common corner are in close proximity to the adjacent bricks. The other two walls have at least one row of empty-studs around them as shown in image (a) of the figure (4.13).

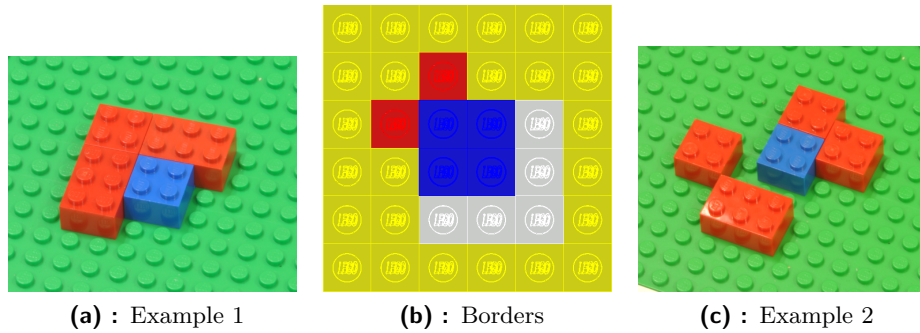


Figure 4.13: Config. 9 - Corner

Image (c) is similar to (c) the images of the Horseshoe (4.3.10) and Box (4.3.11) configurations. Note the differences.

4.3.10 Config 10 - Horseshoe

This configuration concerns that of a horseshoe. One wall of the stone has at least one row of empty-studs around it while the other three walls are in close proximity to the adjacent bricks. See the figure (4.14).

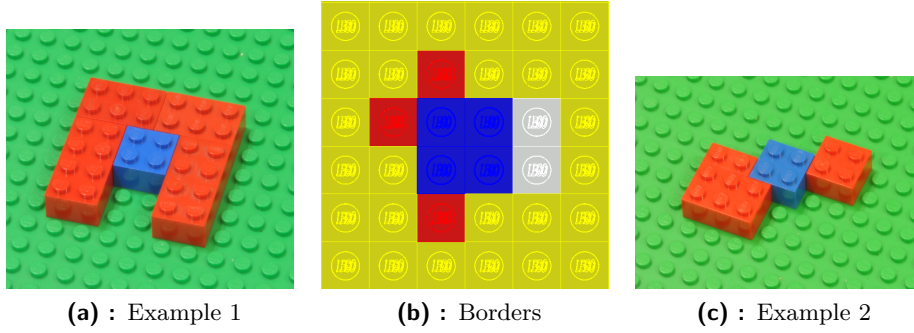


Figure 4.14: Config. 10 - Horseshoe

Image (c) is similar to (c) the images of the Corner (4.3.9) and Box (4.3.11) configurations. Note the differences.

4.3.11 Config 11 - Box

In this configuration, all walls of the stone are in close proximity to its adjacent bricks. It is similar to placing an object into a box. See the figure (4.15). Image (c) shows that adjacent bricks create two inner corners opposite to each other.

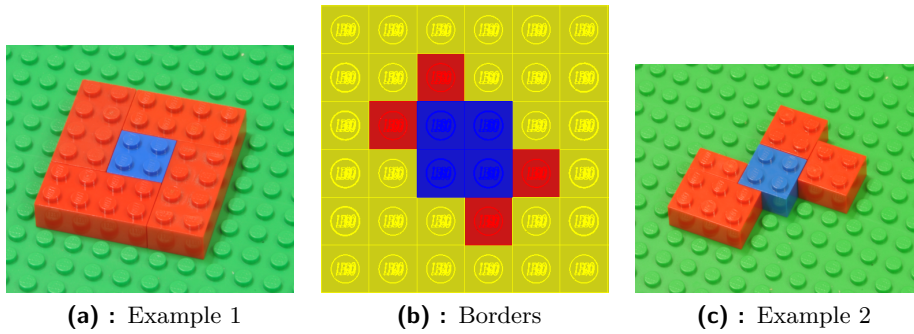


Figure 4.15: Config. 11 - Box

Image (c) is similar to (c) the images of the Horseshoe (4.3.10) and Box (4.3.11) configurations. Note the differences.

This configuration is equal for all four orientations so we do not distinguish between orientations.

4.4 Assembling

This section describes the assembling procedure of LEGO bricks. It is divided into many sub-sections which explain ideas of the proposed movements and discuss parameter setting. The experimental results of each movement are described in chapter (5). By the term assembling procedure it is meant that a situation when the stone is still held by the robot's gripper and it is the aim

to attach the stone into a goal position within the working space. To solve this problem, we have to know which LEGO configuration this situation corresponds to. Based on this knowledge we can then choose the proper movement.

Assembling procedure is very sensitive to precise positioning in all Cartesian axes. The orientation of the stone in the Cartesian space is also important but it is not the critical part, because the robot gives sufficient precision of the stone orientation during the whole manipulation process thus it is not necessary to compensate it.

The most critical part is the measuring of the external force by the unloaded robot. The term unloaded robot means that no external force is acting on the robot. The robot measures always the nonzero influence of the external force and moreover the value is very dependent on the robot's position even if the robot is calibrated to the mass of the gripper. The standard deviation of the measured external force of the unloaded robot in our working space is 3 N. If the robot is close to the kinematical singularity, the inaccuracy of the measurement is much bigger.

All proposed movements use force sensing to be adaptive in the event of an unknown displacement of the real goal position. Motions that use this technique are labeled as force conditioned. These motions are used to attach the stone or check the boundary limits by recording contact with the stone. It is very common that a limit value for the force condition has to be selected from a very narrow range. For these reasons, it is not appropriate to use absolute force value.

To determine the value for a force condition, it is more advisable to use a value that defines the noise. When the noise value is exceeded it means that external force is acting to the robot. The nearby measured external force is taken as an offset. The external force of the unloaded robot is a smooth function in Cartesian space with the low value of its derivative. It is meant that changes of measured external force value are low if the motion is short. Let us refer to the noise force value as **force-limit**.

The principle of external force limit determination for the force conditioned motion is the following. The robot stays in a start position without dynamics. In this position, an external force is measured and the force-limit (noise value) is added. Then, the force conditioned motion is executed. Any force conditioned motion is quite short in order to maintain high resolution of external force. The major advantage of using this approach is the non-destructive handling of LEGO bricks.

The following sections describe proposed movements, whose motions are defined above (4.1). Those motions which are labeled with the same name have also the same purpose but their parameters can differ.

Robot motion is related to a defined coordinate system which is attached to the robot's flange. Detailed information are described in the section (3.1.2).

4.4.1 Trivial movement

This movement is the easiest and the fastest way how to attach the stone into the goal position. It consists of three motions. The *approaching* motion places the stone above the expected goal position. The *attaching* motion places the stone into the goal position. Then the gripper releases the stone and then the *leaving* motion is executed. See the procedure in the figure (4.16).

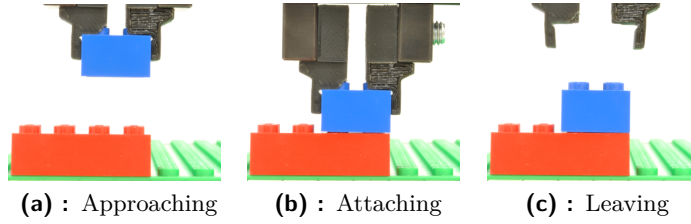


Figure 4.16: Trivial movement phases

This movement is critically sensitive to the Cartesian positioning of the robot. It is possible to use it on well calibrated positions or with a precise robot. This approach is not adaptable in X,Y,Z directions.

With an increasing deviation in X or Y direction, additionally the Z-downwards force to attach the stone increases.

For precise positioning in X and Y, the necessary Z-downwards force is 6 N. For displacement 0.2 mm the force is 65 N and for displacement 0.3 mm the force exceeds 120 N. The stone slips into the final position because of LEGO studs rounding and low stiffness of the robot's adapter.

If Z precision is not sufficient, two situations occur. These effects are nicely observable on the Column configuration (4.3.2). If Z positioning is far high than the reality, there appears a gap between each brick layer and the tower becomes unstable. If Z is too low, the robot smashes bricks. See figure (4.17)

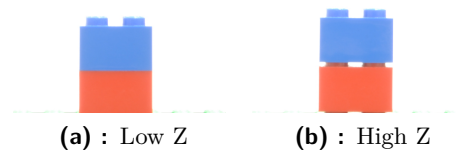


Figure 4.17: Consequences of imprecise Z height

This movement can handle the Standalone (4.3.1), Column (4.3.2), Pathway (4.3.3), Line (4.3.4), Step (4.3.5), Staircase (4.3.6), Neighbour (4.3.7) and Corridor (4.3.8) LEGO configurations.

4.4.2 Tilt movement

This movement is considered in order to replace the Trivial solution (4.4.1). It is applicable for those configurations which are not in close proximity to

adjacent bricks due to deviations in X and Y directions. This movement can adapt to small displacements in X, Y and Z directions. It consists of seven motions. The end positions of these motions are captured in the figure (4.18).

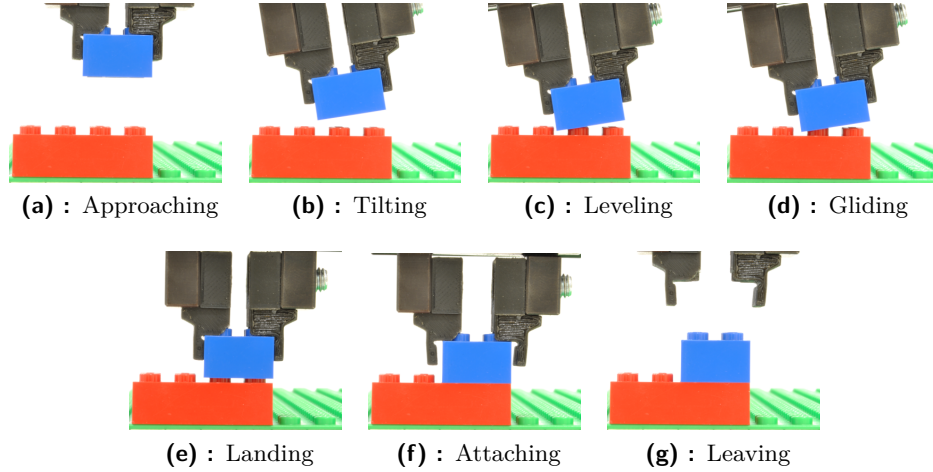


Figure 4.18: Tilt movement phases

After the *approaching* motion, the *tilting* motion is executed. It rotates the stone to prepare it for the *leveling* motion. This tilting causes the one margin-edge to become the lowest part of the stone in Z. The stone is simultaneously placed into the goal layer 1 mm above the brickwork's stud-top. The angle of rotation is determined according to figure (4.19), where the highest margin-edge is approximately 2σ above the brickwork's stud-top. It corresponds to 10° . Either way, the angle can be chosen from a quite large range.

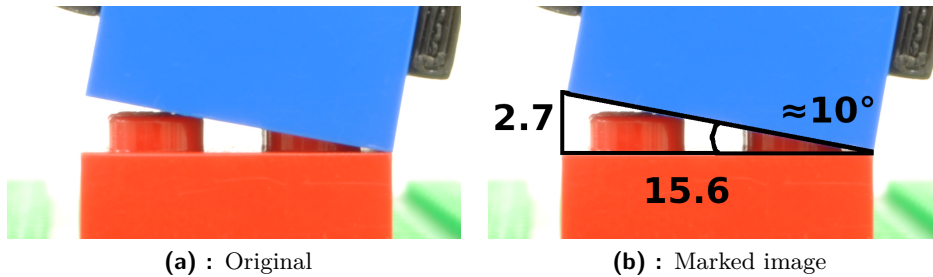


Figure 4.19: Angle of the second motion and marked hit contact of the fifth motion

The *leveling* motion decreases the inaccuracy of the real Z position of the brickwork. The motion is stopped when the stone hits the brickwork. This motion is sensitive in X and Y deviations. If the X and Y are highly deviated, the *leveling* motion wrongly determines the Z height. This will result in the margin-edge or the stone's skewed bottom-area hitting the brickwork's stud-top. Then, the following motion misses the real goal position.

If the X and Y deviations are in limits, the margin-edge is placed between

brickwork's studs and contact occurs between stud-top and stone's tube-edge. The set force-limit in Z is to 4 N. The limit for X and Y deviation is up to 1 mm.

Only for the case of *gliding* motion, let's label the direction towards the goal position as X and the perpendicular direction as Y . The *gliding* motion is executed and stopped when the padding-edge hits the brickwork's stud-walls. The force-limit is set to 4 N in X . The stiffness and damping ratio of translations and rotations are set to maximal values to keep its invariant, except Y direction which has its damping ratio set to a minimal value of 0.1 and the stiffness to 1.5 N. This extremely loosened setting causes the stone's adaptation in the Y direction when contact occurs. The X direction is adapted by the force condition. The result of the *gliding* motion is that the padding-edge is held by the brickwork's stud-walls.

The *landing* motion aligns the stone onto brickwork studs. The stone is situated into the goal X and Y position.

The *attaching* motion places the stone Z-downwards until the force-limit is exceeded. The force-limit is set to 13 N. The lower force-limit causes unpropped Z placing while the higher force-limit causes the detaching of the non-solid brickwork.

The gripper is opened and the *leaving* motion is executed.

This movement can handle the Standalone (4.3.1), Column (4.3.2), Step (4.3.5), Staircase (4.3.6) LEGO configuration. If the gripper is correctly oriented, as shown in the figure (4.20), and the low deviation in a specific direction has occurred, it is possible to handle the Pathway (4.3.3), Line (4.3.4), Neighbour (4.3.7) and Corridor (4.3.8) LEGO configurations.

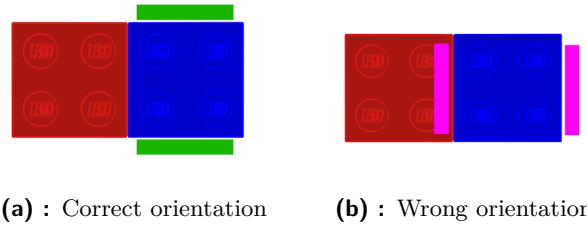


Figure 4.20: The correctly and wrongly oriented gripper when attaching the stone into the goal position which is situated in close proximity to adjacent brick.

4.4.3 Spiral movement

This movement is based on a flat spiral. The flat spiral is executed in the XY plane. It is proposed to cover a high position deviation. The spiral function is predefined in the robot's software environment. This movement is adaptable in all directions. This movement consists of six motions as shown in the figure (4.21).

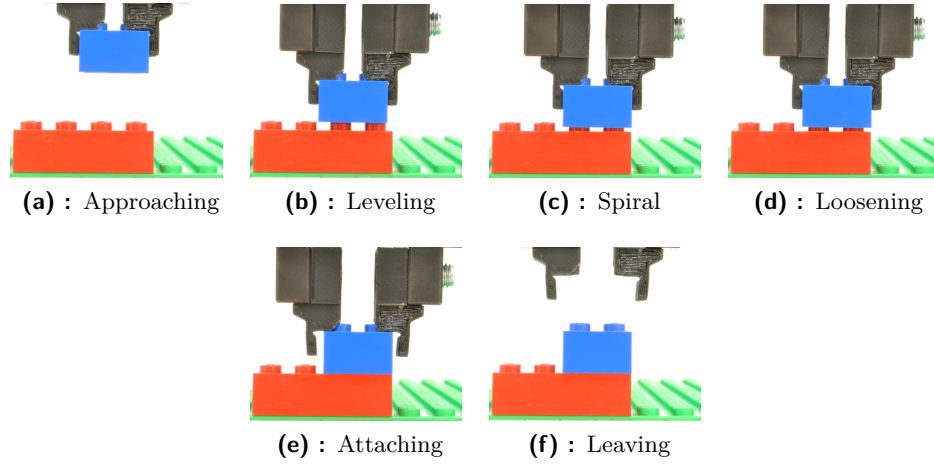


Figure 4.21: Spiral movement phases

After the *approaching*, the *leveling* motion is executed. It moves the stone Z-downwards until the force-limit 10 N is not exceeded. In the case of low deviation in X and Y directions, the stone is placed directly onto the brickwork's studs and the *spiral* motion has a negligible effect. In the case of high deviation in X and Y directions, the motion is interrupted in a position, where the bottom-area comes in contact with brickwork's stud-tops. The external force still affects the stone even after the motion interruption. The force-limit can be higher, but gliding of bricks in the next motion is not so soft.

It is highly probable that the stone does not end in the goal position as in image (b). For that reason, the *spiral* motion is executed. This motion adapts the stone's position into the real X and Y goal position.

The inbuilt spiral function has several parameters like frequency, amplitude, stiffness and duration time. The amplitude is chosen experimentally as 20 mm because it is very affected by the compliant parameters and it does not correspond to the real values. The real amplitude is useful up to 4 mm because it is the maximal deviation in X and Y direction to prevent position ambiguity.

The tuning of the frequency goes together with the duration time. Generally, it is an effort to make a very fine spiral with many spins in order to prevent missing the real goal position. If we are not concerned about the execution speed of the motion, the best option is to set a low frequency and an extremely long duration time. The frequency is set to 4 Hz and the duration time to 12 s. This frequency value seems to be critical high. In the deviation of more than 3 mm, the stone loses an ability to stop in the real goal position and slides over it. In the case of having this frequency and deviation of 2 mm in X and Y direction, the spiral finishes up to 6 s due to a force-limit.

The duration time of the spiral is the sum of the rising time, hold time and fall time. We set the rise time as 90% of the duration time because it determines the changing of the spiral spin diameter. The fall time is set to 10% of the duration time, while the hold time is zero. The fall time is nonzero

to return the brick back to the start position if the force condition is not violated.

The stiffness is set to 4000 N/m for all directions. This stiffness setting ensures that the spiral motion continues even if the stone comes into contact with an adjacent brick and additionally this adjacent brick is not detached by the stone's force acting.

That was a description of the setting of spiral parameters and their consequences. The following part describes compliant parameter setting.

The damping is set to the maximum value for all degrees of freedom (DOFs) which ensures that in the case of stone deviation due to external forcing, the stone immediately returns to its original position after this external force passes.

The bias in the Z direction is set to 40 N which provides continuous Z-downwards pressure. This value is also set experimentally. When the stone finds the real X and Y goal position, this pressure force is enough to push the stone onto the brickwork's studs.

To prevent damage of the gripper or the stone by increasing a spin radius, a force-limit in the XY plane to 12 N has been set.

When this spiral motion with its described setting is executed, then the stone is placed in the real X and Y goal position. However, there is a presence of the external force caused by the high force-limit. This external force causes tension between the stone and the adjacent bricks. For that reason, the fast *loosening* motion is executed. Its duration is 0.7s and the compliant parameters are set as 0.8 N for translation stiffness, the maximal value for rotation stiffness and 0.3 for damping ratio of all DOFs.

The *attaching* motion places the stone Z-downwards into the real goal position. Then the gripper is released.

Then the *leaving* motion is executed.

This movement can handle the Standalone (4.3.1), Column (4.3.2) and Pathway (4.3.3) LEGO configuration. If the gripper is correctly oriented, as shown in the figure (4.20), and the low deviation in a specific direction has occurred, it is possible to handle the Line (4.3.4), Neighbour (4.3.7) and Corridor (4.3.8) LEGO configurations. If this movement uses two spiral motions instead of only one, this movement can handle the Corridor (4.3.8) configuration without previous deviation restriction. See Experiments (5).

■ 4.4.4 Neighbour movement

This movement is designed for the Neighbour (4.3.7) LEGO configuration. It combines a touching part and the Tilt movement (4.4.2). This motion uses the adjacent brick to reduce deviation of the expected goal position. It consists of 10 motions as shown in the figure (4.22).

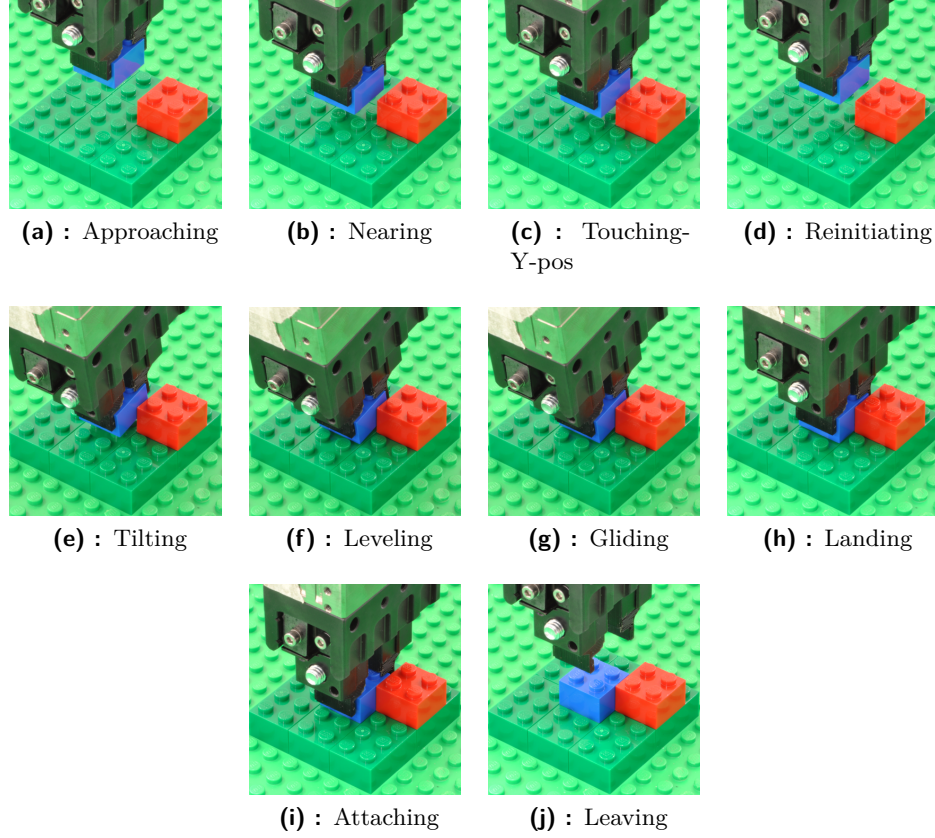


Figure 4.22: Neighbour movement phases

In the same manner as for the *gliding* motion in Tilt movement (4.4.2), let's label the direction towards the goal position from the adjacent brick position as *Y-negative* direction, the backward direction as *Y-positive* and the perpendicular direction as *X*. After the *approaching* motion, the *nearing* motion is executed. The stone is placed at a collision height such that the stone's bottom-area is lower than the top edge of the adjacent brick. The stone's bottom-edge is placed 4 mm below the top edge of the adjacent brick. Moreover, the stone is deviated 4 mm in *Y-negative* direction in order to prevent a collision with the adjacent brick in the case of a high *Y* deviation.

The *touching-Y-pos* motion moves the stone in *Y-positive* direction until the leading wall does not come into contact with the wall of the adjacent brick and the force-limit is exceeded. The force limit is set to 4 N. This value can be relatively high because quite a large area comes into contact, thus the force is not transferred just over one point but over an area. When the *touching* motion is interrupted, the stone position is taken as the reference for the real *Y* goal position.

The *reinitiating* motion places the stone above the new expected goal position in the same *Z* height as the *approaching* motion. The *Y* position of the new expected goal position is the real *Y* goal position. This position is computed as being the contact position of the stone shifted by the LEGO

gap between bricks in the Y-negative direction. The expected goal position is still imprecise in Z and X directions which the following motions eliminate.

The following motions are in the fact the Tilt movement (4.4.2), whose stone's end position of the *approaching* motion is equal to the stone's end position of the *reinitiating* motion.

4.4.5 Corner movement

This movement is designed for Corner (4.3.9) LEGO configuration, where the inner corner from the adjacent bricks is made. This movement uses the adjacent bricks as reference positions in order to determine the real goal position. This movement works on the techniques used in the Neighbour (4.4.4) movement. This movement can handle big deviations in the XY plane. It consists of 13 motions as shown in the figure (4.23).

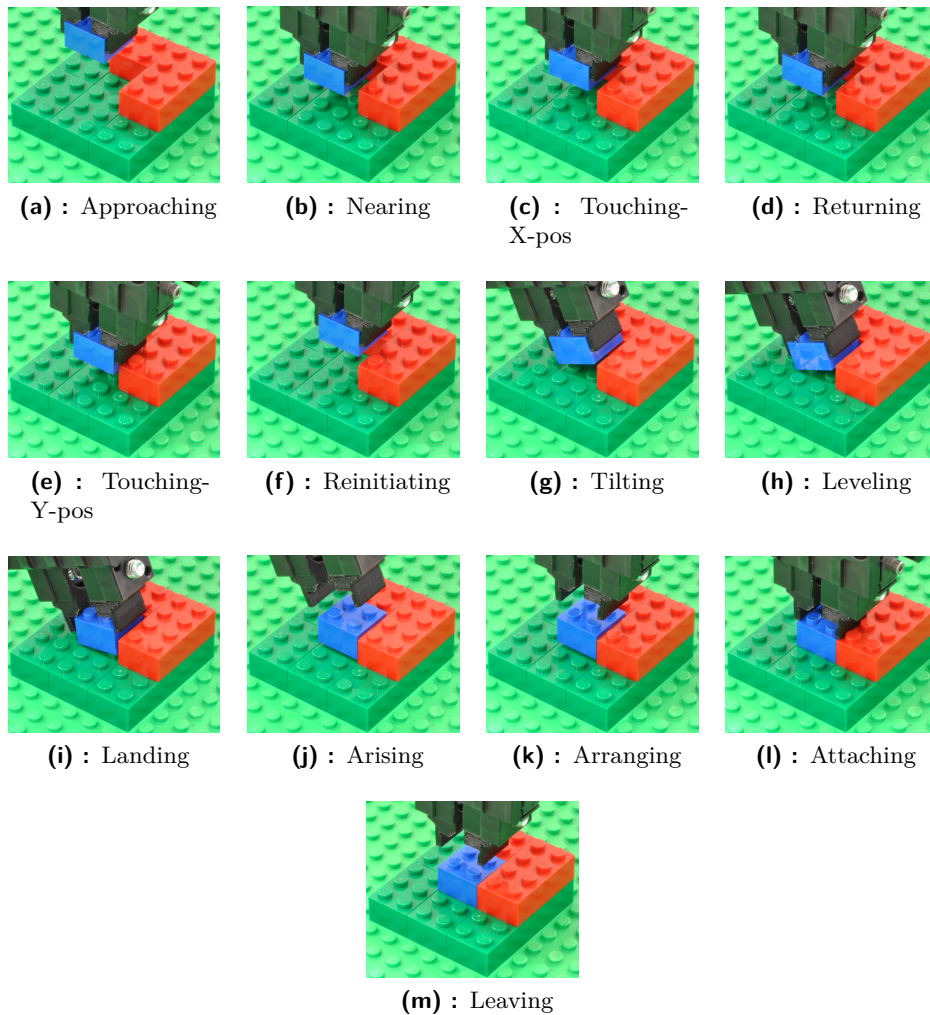


Figure 4.23: Corner movement phases

Let's keep the labeled axes from the Neighbour (4.4.4) movement as the

images are equally composed. Thus the direction from the goal position to the adjacent brick placed closest to the image (a) bottom is *Y-positive* the backward direction is *Y-negative*. The direction from the goal position to the adjacent brick placed closest to the image (a) top is *X-positive* the backward direction is *X-negative*. Thus the direction from the goal position to the inner corner is *X-positive*, *Y-positive*. Images (c) and (e) are named according to their motion direction.

After the *approaching* motion, the *nearing* motion is executed. The setting of this motion is similar to that of the Neighbour (4.4.4) movement. The nearing position deviated 4 mm in *X-negative* and *Y-negative* directions from the expected goal position to prevent collision of the stone and any adjacent brick. The height of the bottom-area is 4 mm below the top-area of the adjacent bricks.

The *touching-X-pos* motion is executed. Its end position serves as a reference for the real X goal position. The force-limit is 4 N.

Then *returning* motion is executed in order to avoid friction of previous contact position. This motion can be omitted because the friction level is very low. It better distinguishes the touching motions in the images. The *touching-Y-pos* motion is executed and the reference Y position is gained. The force-limit is the same as for *touching-X-pos*. The new expected goal position is computed. The position is imprecise only in the Z direction. The real X and Y positions are gained by adding a LEGO gap, which is 0.2 mm, in the *X-negative* and *Y-negative* directions.

The *reinitating* motion is executed and the stone is placed above the new expected goal position. The *tilting* motion is executed. The stone is tilted around the padding-edge of the wall which is not in close proximity to the adjacent brick in order to prevent a collision. The tilting angle is wider than in the Tilt (4.4.2) movement. The angle is 25° so as to keep the opening of the gripper feasible. The Z height is the same as in the Tilt (4.4.2) movement's motion.

Then the *leveling* motion with its force-limit set to 4 N is executed. The force-limit is low in order to prevent causing tension to the stone.

When the margin-edge comes into contact with the brickwork, the gripper releases the stone which falls onto a brickwork's studs. By this opening action, the *landing* motion is performed. Then, the *arising* motion is slowly executed in order to not deviate the stone from its position.

The *arranging* motion consists of aligning the gripper orientation, shifting the gripper in the XY plane to get above the goal position and then closing the gripper.

The *attaching* of the stone into the goal position is achieved through finger pressure. The force-limit is set to 13 N which establishes the proper attaching.

The brick is placed into the goal position. And the *leaving* motion is executed.

4.4.6 Horseshoe motion

This movement is designed for the Horseshoe (4.3.10) LEGO configuration, where the adjacent bricks are in close proximity to three walls of the stone. But this movement is applicable only if the stone's wall has at least three rows of empty-studs around. The movement consists of 14 motions as shown in the figure (4.24).

The idea of this movement is simple. The brick is placed by the *spiral* motion into the horseshoe valley and the goal position is reached by finger handling. This motion can handle only very low goal position deviation.

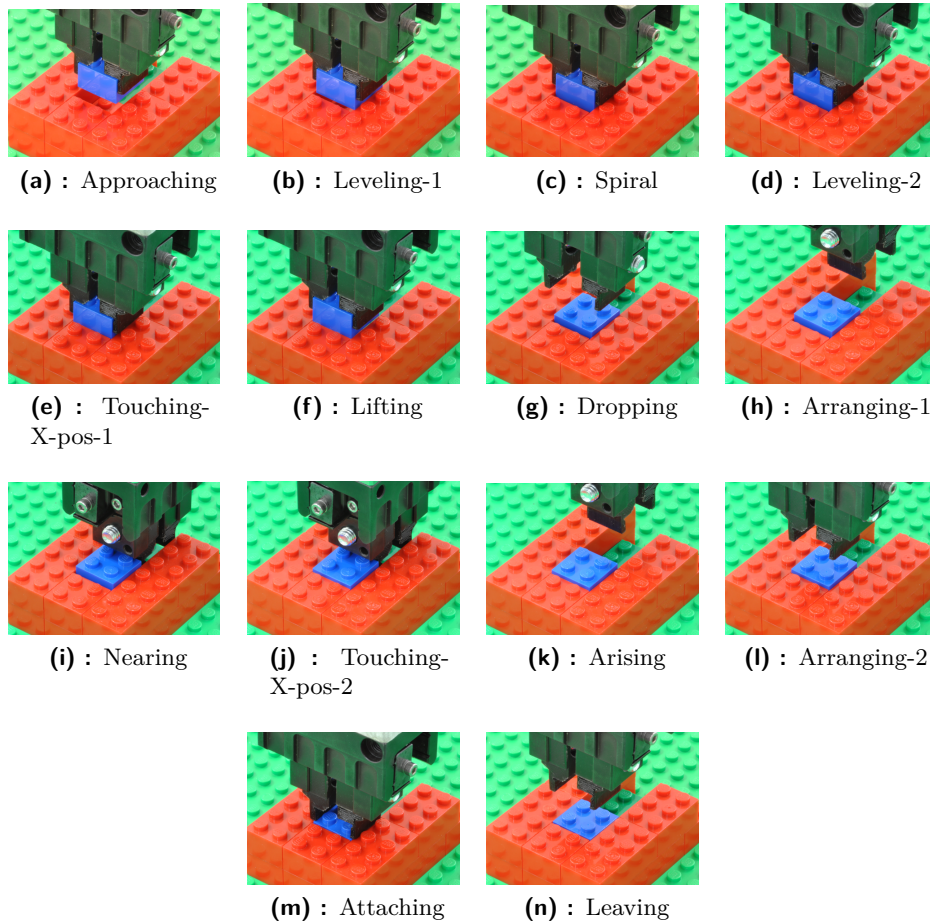


Figure 4.24: Horseshoe movement phases

Let's label the direction axes in XY plane for this movement. Let's call the direction from the goal position towards the empty-studs the *X-negative* and the backward direction the *X-positive*. The perpendicular direction is labeled simply as *Y*.

The *approaching* motion places the stone above its expected goal position.

During the execution of the *leveling-1* motion, three cases can occur. The most favourable case is when the stone ends in the horseshoe's valley. It means

that it would be possible to skip the *spiral* motion. This case occurs when the *Y* position is precise and the *X* position is deviated in the *X-negative* direction. The second case occurs when the bottom-area comes into contact with the top-area of the adjacent bricks. The third case occurs when the bottom-area comes into contact with the stud-tops of the adjacent bricks. The last two cases mostly fail the movement due to the set force condition in the spiral motion. The most obvious fail position is when the second case occurs, where the stone is into contact with the stud-walls of the two adjacent bricks. In fact, it is in a corner surrounded by stud-walls. In this fail position the force condition is fired before the stone is placed into the valley.

The *spiral* motion is the most critical part. The reason is described above. The setting is equal to that of the Spiral (4.4.3) movement. If this motion is executed successfully, the stone's bottom-area is placed into the valley.

The *leveling-2* motion moves the stone Z-downwards in order to minimize the risk of slipping the stone out of the valley during the following motions. The force-limit is set to 6 N.

The *touching-X-pos-1* motion places the stone closer to the real X goal position. The Y position is precise because the brick is surrounded in both Y directions and the beveling of studs ensure the right positioning. The force-limit is set to 10 N.

During the *lifting* motion, the stone is moved 0.2 mm Z-upwards in order to prevent damage of the fingers during opening.

Then the gripper is opened and the stone drops onto the brickwork's studs which fulfils the *dropping* motion. It is necessary to adjust the X position of the stone, which is performed during the following four motions.

The *arranging-1* motion closes the gripper, orients it and places it into a non-collision height above the empty-studs.

The *nearing* motion places the fingers into the valley. Then, the *touching-X-pos-2* motion adjusts the stone into the real X goal position. The force-limit is 4 N.

The *arising* motion is slowly executed and thereby the stone position is sustained by the sliding of the finger's wall over the stone's wall.

The *arranging-2* motion orients the gripper and places it above the real goal position.

The *attaching* motion moves the gripper Z-downwards and thereby the stone is attached. The force-limit is 13 N.

The *leaving* motion finalizes this movement.

4.4.7 Box-spiral movement

This movement is used for the assembling of the Box (4.3.11) LEGO configuration as well as the Brox-corner (4.4.8) movement. It consists of nine motions as shown in the figure (4.25). The figure (4.26) shows images with the one skewed and partially transparent brick. The purpose of the brick is to make the stone's margin-edge positions visible in the box area in all phases of this movement.

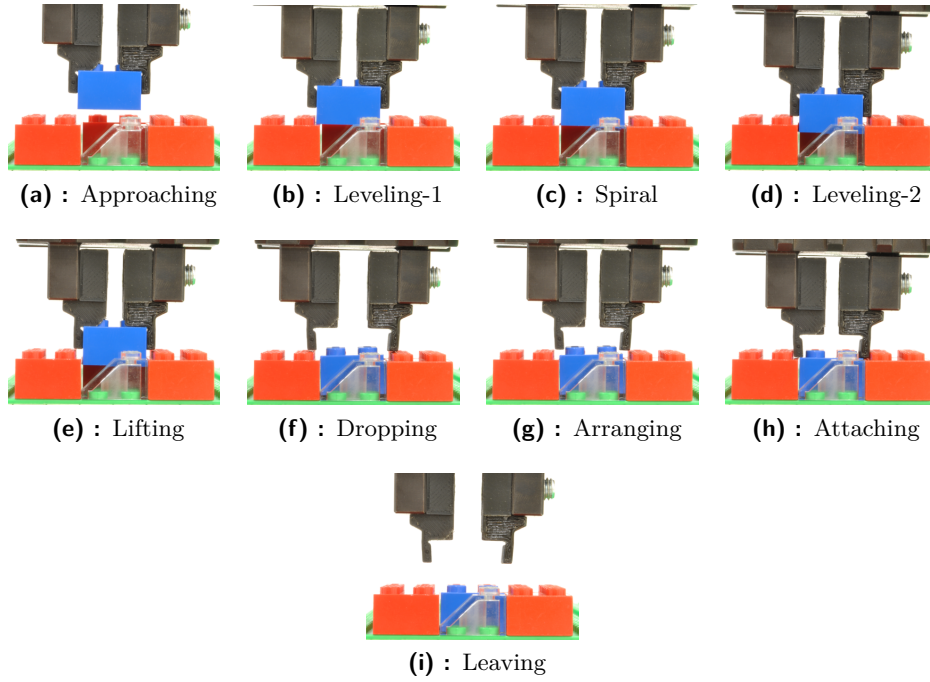


Figure 4.25: Box-spiral movement phases. In images, there is a transparent skewed brick which enables a position of the stone being visible in each motion.

In image (a), where the *approaching* position is depicted, it is visible that the expected goal position is deviated from the real one.

The force-limit of the *leveling-1* motion is set to 1.5 N. At the end of this motion, it is not necessary to stop the stone in a contact position because the next spiral motion applies a Z-downward force to keep pressure. This motion ends up in four positions. The three positions were described in the Spiral (4.4.3) movement in the *leveling* motion. The fourth case represents the situation when the stone stops in mid-air due to the high noise level of the measured external force.

In the case with the reasonable position deviation, the stone has ended in the contact position with the top-area of the adjacent brick as shown in the image (b).

The *spiral* motion is executed with equal setting as in the Spiral (4.4.3) movement.

The *leveling-2* motion serves to get the stone closer to the goal position so as to not fall out during the *dropping* motion. The force-limit is set to 6 N.

The Z-downwards pressure force from the previous motion is eliminated by the *lifting* motion which moves the brick 0.5 mm Z-upwards to avoid finger damage during opening.

The *dropping* motion is executed and the stone falls into the goal position. Because the brick is bordered in the XY plane, it remains to position the stone in the Z direction.

The *arranging* motion closes the gripper above the real goal position and the *attaching* motion with the Z-downwards force-limit set to 13 N attaches

the brick into the goal position.

During the *leaving* motion the gripper reaches the non-collision height.

4.4.8 Box-corner movement

This movement is designed for the Box (4.3.11) LEGO configuration as the previous Box-Spiral (4.4.7) movement. Moreover, it is usable for the Horseshoe (4.3.10) and the Corner (4.3.9) LEGO configurations, but only for those orientations where the inner corner is created from the adjacent bricks. This movement consists of nine motions as shown in the figure (4.26). This motion can adapt the stone to a higher position deviation than the Box-Spiral (4.4.7) movement.

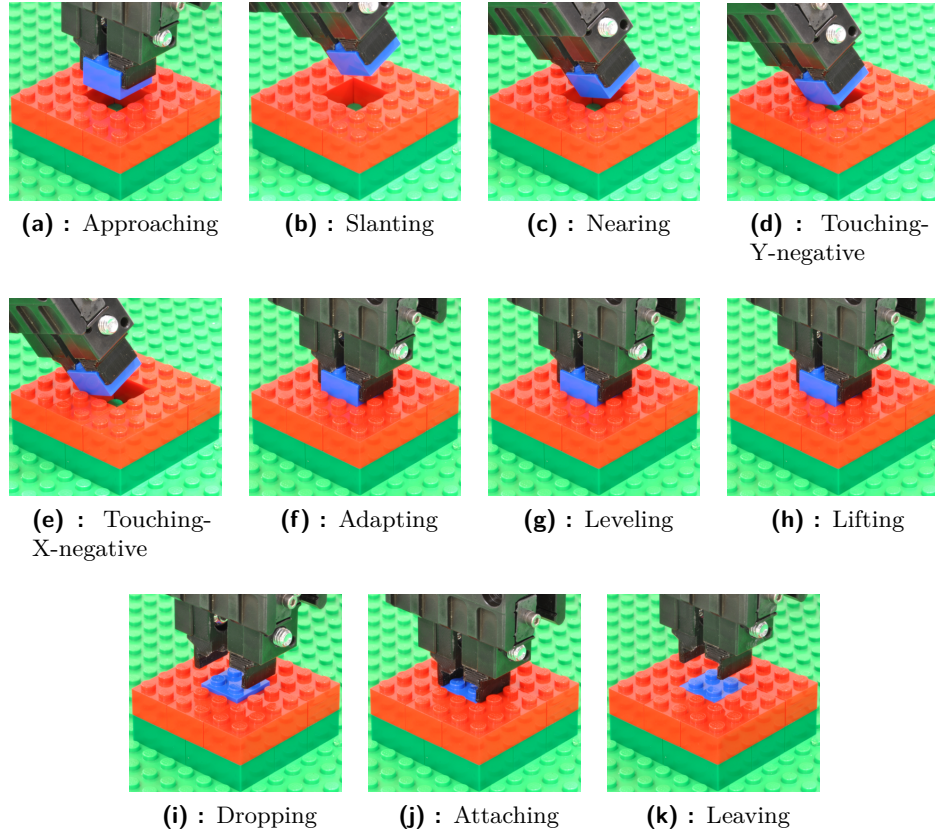


Figure 4.26: Box-corner movement phases

Let's label the X and Y axes according to the images (d) and (e), where the Y direction is oriented to the direction of the left upper corner of the image.

After the *approaching* motion, the *slanting* motion is executed. The stone is rotated $\pi/8$ in x and y axis of the padding-edge which corresponds that the stone is rotated around the static bottom padding corner. Let's call this corner the pointing corner. The pointing corner becomes the lowest part of

the stone. Simultaneously, the stone is shifted $\frac{1}{4}$ of the stone's width in the *X-positive* and *Y-positive* directions. Also, at the same time, the stone is lifted 4 mm Z-upwards to keep the non-collision height.

The angle of the rotations is equal because no one orientation axis is privileged or more important. The angle can be arbitrarily selected. It determines the contact area in the following *touching* motions. The lower angle ensures better contact due to the stone's corner not being so skewed, while the higher angle ensures a higher distance between the stone's bottom-area and the top-areas of the adjacent bricks.

The stone's shifting of $\frac{1}{4}$ of the stone's width is selected according to the maximal permissible brick deviation.

The *nearing* motion places the pointing corner approximately 3 mm below the top-area of the adjacent bricks.

Then the *touching-Y-negative* and *touching-X-negative* motions are executed in order to place the pointing corner as close to its real goal position as possible. Simultaneously, the stone is blocked in the inner corner to be movable in the *X-negative* and *Y-negative* directions. The force-limit is set to 4 N for both motions.

As shown in image (e) the stone is rotated and placed in the inner corner. The *adapting* motion aligns the stone in compliant mode. The aim of this motion is to place the stone above the real goal position. The motion is slowly executed so that all forces can influence the stone's position. It takes 4 s. The motion is 3 N biased in Z-downwards to ensure that the stone does not slipping out. The stiffness is set to 300 Nm/rad for all rotations, which is the maximal value, and 4 N/m for all translations to keep position instability. The damping is set to the maximal value in order to overcome friction. At the end of the *adapting* motion, the stone's bottom-area is placed in a collision Z height and thus the stone is bordered by the adjacent bricks.

The *leveling* motion is executed with the set force-limit to 6 N. Then, the *lifting* motion places the stone 0.5 mm above to prevent the fingers being damage and by applying the *dropping* motion the stone is dropped onto the studs of the brickwork.

The gripper is closed and *attaching* motion moves the gripper Z-downwards until the force-limit do not exceed 17 N.

The stone is attached into the goal position by fingers and the *leaving* motion lifts the gripper into the non-collision Z height.

4.5 Disassembling

This section describes the disassembling procedure of LEGO bricks. It is divided into three sub-sections. By the term disassembling procedure, it is meant a situation when the gripper is not gripping any brick yet and the distance between its fingers is at least the stone's width in order for it to be able to fit one there. In our implementation, it is approximately 17 mm.

The disassembling procedure has different problems than the assembling procedure. We are not able to disassemble all the aforementioned configura-

tions. The limitations are in the concept of our fingers. We can only detach a stone if it has empty-studs around in appropriate positions.

The disassembling procedure is sensitive to a brick's precise positioning. Our implementation of the disassembling movements can handle lower deviations of real positions than some of the proposed assembling movements. The reason lies in fingers construction. Collisions occur in two cases. The first collision case is when a finger-tip hits a top-area of the stone. The second collision case is when a finger hits the stud-top of the stone.

The two major problems of the disassembling procedure are in the slipping of the stone from the fingers and in the detaching of multiple bricks at once.

The slipping of the stone is shown in the figure (4.27). It happens when the gripper executes the *tilting* motion and the real rotation axis is misplaced. Another reason for brick slipping is the low friction surface of the gripping fingers.

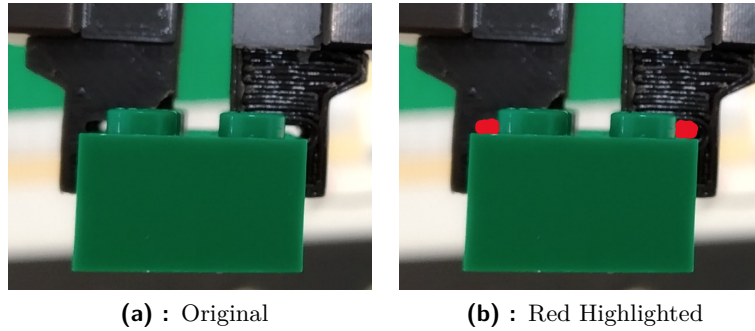


Figure 4.27: The slip of the brick during the detachment of a stone

The detaching of multiple bricks solves motions (4.5.2) and especially the (4.5.3) which were conceived for that very reason.

The LEGO configurations which we can disassemble are Standalone (4.3.1), Column (4.3.2), Pathway (4.3.3), Line (4.3.4), Step (4.3.5), Staircase (4.3.6), Neighbour (4.3.7) and Corridor (4.3.8) because in these configurations we can place the fingers onto the stone.

■ 4.5.1 Trivial movement

This motion is the simplest one. It consists of four motions as shown in the figure (4.28). This motion doesn't use any compliant mode or breaking force conditions. It is possible to use these techniques, although it doesn't solve the problem of the detaching of multiple bricks.

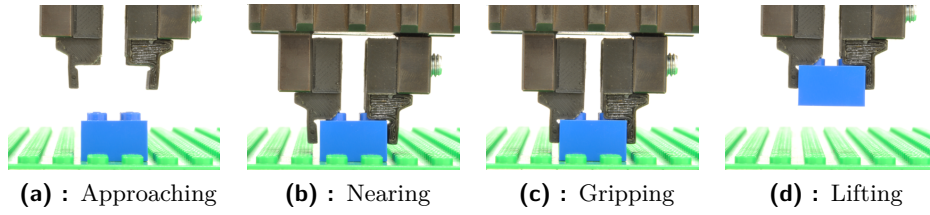


Figure 4.28: Trivial movement phases

After the *approaching* motion, the *nearing* motion is executed in Z-downwards. The expected Z goal position has to be higher or equal to the real goal position to grip the stone in the *gripping* motion.

The *lifting* motion lifts the stone, mostly with other underlying bricks of the brickwork, up into non-collision Z height.

This movement is prone to detaching more bricks instead of only the stone intended. It may happen that the brickwork is partially detached during the movement.

This movement is successful in Step (4.3.5) LEGO configuration. If the brickwork underneath the stone is composed of larger bricks than the stone and the gripper is appropriately oriented, then the Standalone (4.3.1), Staircase (4.3.6), Pathway (4.3.3), Neighbour (4.3.7) and Corridor (4.3.8) LEGO configurations are also successfully manageable.

If the fingers can grip the Line (4.3.4) LEGO configuration then it is also possible to handle it.

4.5.2 Tilt movement

This movement is considered to replace the Trivial (4.5.1) solution. It is inspired by the human manipulation. This movement consists of seven motions as shown in the figure (4.29).

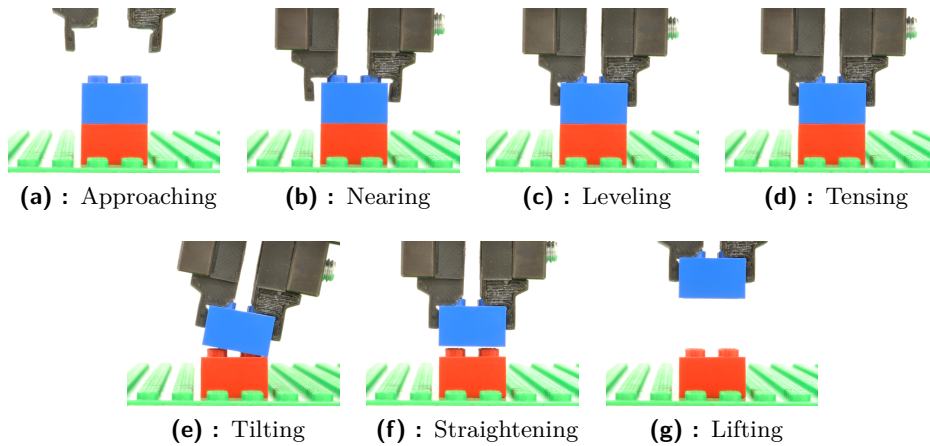


Figure 4.29: Phases of the Tilt movement

Let's label the direction from right to left as *X-positive* according to the arbitrary image from the figure (4.5.2) and the opposite direction as *X-negative*.

Like the first, the *approaching* motion is executed and the gripper opened.

The *nearing* motion gets the gripper 1 mm above the expected goal position. The Z height can be changed to overcome higher deviation.

The *leveling* motion serves to reveal the real Z height of the stone. The force-limit is set to 1.5 N. Because the expected motion trajectory is very short, the force-limit can be very low. The motion is interrupted when the fingers come into contact with the top-area of the stone. Then the gripper is closed as shown in image (c).

The *tensing* motion shifts the gripped and still attached stone 0.1 mm Z-upwards and 0.4 mm in the *X-positive* direction. It detaches the stone a little bit from the brickwork and prevents against the unintentional detachment of multiple bricks.

The *tilting* motion partially detaches the stone. The angle is equal, as in the Tilt movement (4.4.2). Moreover, the stone is lifted 0.4 mm Z-upwards and shifted 0.4 mm in the *X-negative* direction to compensate for the shift from the *tensing* motion.

The *straightening* motion aligns and completely detaches the stone.

The *lifting* motion places the stone into the non-collision Z-height.

■ 4.5.3 Tilt-Repairing movement

This movement is the compliant variant of the previous one (4.5.2). It is enhanced by the repairing part. It was originally designed for a specific brickwork layout (4.31) which is extremely prone to detaching multiple bricks. This movement consists of 11 motions as shown in the figure (4.30).

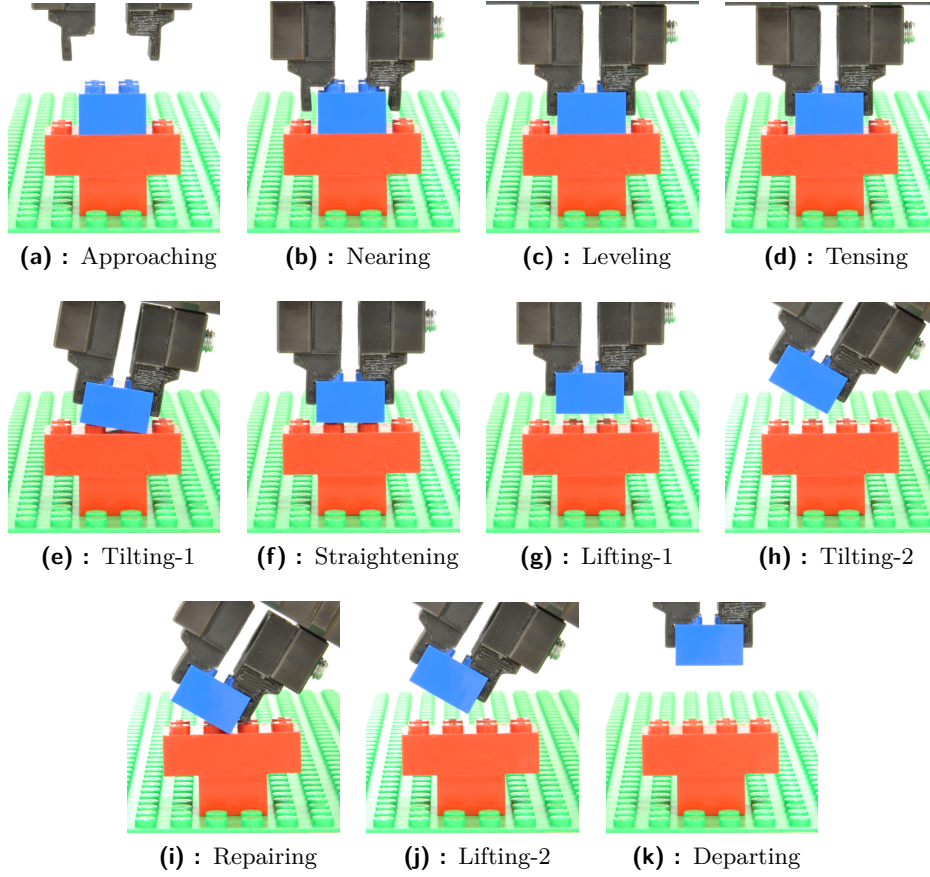


Figure 4.30: Phases of the Tilt-Repairing movement

Let's label the direction from right to left as *X-positive* and the opposite direction as *X-negative* according to the Tilt (4.5.2) movement.

The *approaching*, the *nearing* and the *leveling* motions are equal as in the Tilt (4.5.2) movement. Which means that at the end of the *leveling* motion, the fingers are still holding the attached stone without any forces act on the stone.

The *tensing* motion has only been set in a compliant mode. The end position is set as the start position but the compliant mode overlays it and thus it changes the position a little. The acting coordinate system is placed in the padding-edge of the stone. The stiffness for all translation axes is set to 3000 N/m in order to be looser than the Y axis which is set to the maximal value. The position in the Y axis is constant during the whole movement.

The stiffness of all rotation axes is set to maximal value 300 Nm/rad. The damping ratio is also set to the maximal value so as to retain the gripping position.

The most important is the stretching force vector which is shown in the figure (4.31). The vector starts in the rotation axis of the stone and ends at the margin-edge of the underneath brick. The force is acting on the underneath brick so as to not tilt it simultaneously with the stone in the next

motion. The stretching force vector is composed of Z-downwards force vector and *X-positive* force vector. The force ratio is $X : Z = 17.8 : 9.6 = 9 : 5$ and it proceeds from the 2x2 LEGO brick dimensions. The *X-positive* force is set to 9 N and the Z-downwards force to 5 N.

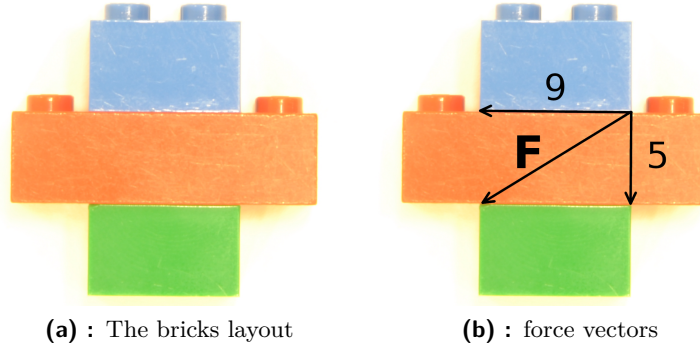


Figure 4.31: The stretching force vector (\mathbf{F}) of the *tensing* motion

The *tensing* motion lasts 300 ms which is enough to reflect the force vector on the stone and the brickwork.

The *tilting-1* and the *straightening* motions are equal as in the Tilt (4.5.2) movement.

The *lifting-1* motion moves the aligned stone 2 mm Z-upwards to prevent collision during the next motion.

The *tilting-2* motion rotates the stone to create the pointing margin-edge as shown in image (h). The inclination angle is $\pi/6$. A higher angle could cause a collision of the finger with the top-studs of the brickwork during the next motion.

The *repairing* motion has been set to the force-limit 15 N in the Z direction in order to align the brickwork.

The brickwork is aligned and the *lifting-2* motion lifts the stone up into the end position of the *tilting-2* motion.

The *departing* motion aligns the stone and lifts it into the non-collision Z height.

Chapter 5

Experiments

This chapter presents the experimental results of the chosen configurations. The experiments present the success rates of the proposed movements handling the selected configuration when a position deviation occurs. The following figures always consist of at least two images. Every figure represents a LEGO configuration. Its images represent the tested movements and, if necessary, the orientation image defining the axes of orientation for the figure. The tested LEGO configurations are always the Examples-1 from the chapter (4.3). The deviation is tested in the XY plane. The Z direction is not tested. The reason for this is that besides the Trivial movements, every other movement is endowed with the *leveling* motion which covers the imprecision in the Z direction and ensures the correct grip height for the stone. Without a precise grip height, every movement would fail.

All images have the same range in the X and the Y directions. The range is $<-4, 4>$ mm and the step is 0.2 mm. There are three colours in the images. The green colour represents deviations which are successfully handled. It means that the goal position is reached with high repeatability. The dark green colour represents those deviations which are partially successful. These deviations cover all cases which are not correctly handled e.g. the brickwork being partially detached, the stone being attached into the wrong position, the stone not being attached properly and sometimes the manipulation being completely failed. The red colour represents deviations which are always unsuccessful.

In this chapter, all proposed movements have been tested at least once. Here are not present all combinations of the proposed movements and LEGO configurations but the presented experiments have been selected to cover most of the results and others can be easily deduced from these.

The areas in the resulting images are very often neither symmetric nor centred. The reasons lie in a wrongly determined coordinate system for the gripper and the dependence of the measured external force on the robot's position.

All the tested movements are set according to their description in chapters (4.4) and (4.5). Let's notice that the biggest reasonable deviation of LEGO bricks is 4 mm in the X and Y direction.

Assembling Experiments

The figure (5.1) shows that Tilt movements are more durable than the Trivial movement and the Spiral movement can withstand the biggest deviations.

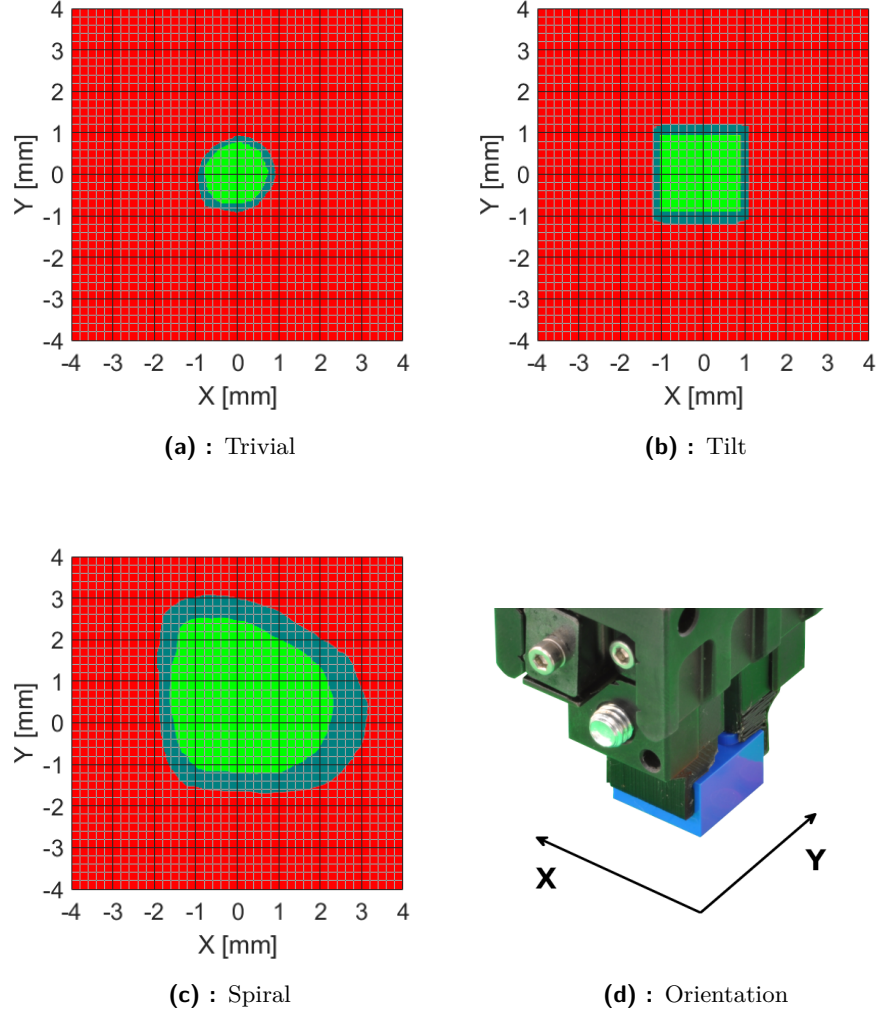


Figure 5.1: Column configuration

The figure (5.2) shows only the Trivial movement for the Line configuration. In image (a), there are visible niches which represent the position, where the stone comes into collision with the adjacent bricks. In the case of performing the Tilt movement, it would look similar. The image would preserve the niches in the same positions but the rest space would be as large as in the figure (5.1 b).

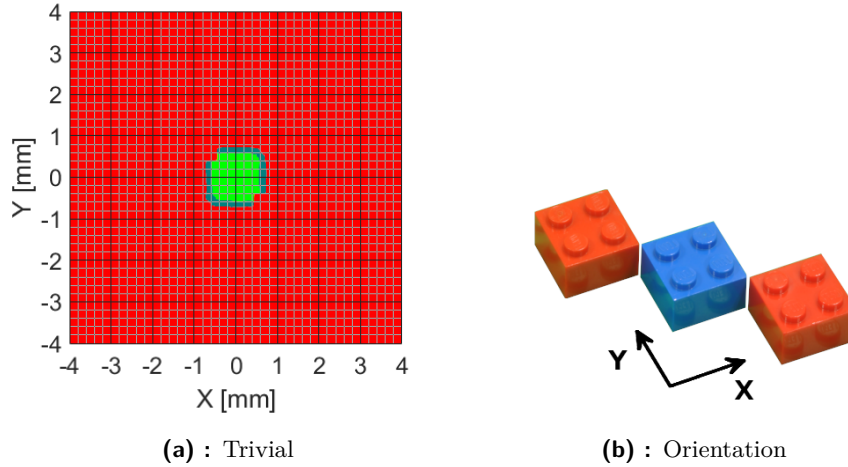


Figure 5.2: Line configuration

The figure (5.3) shows the Neighbour configuration being handled by its designed Neighbour movement. Due to the *Touching* motion, almost the whole range of the X axis can be successfully handled.

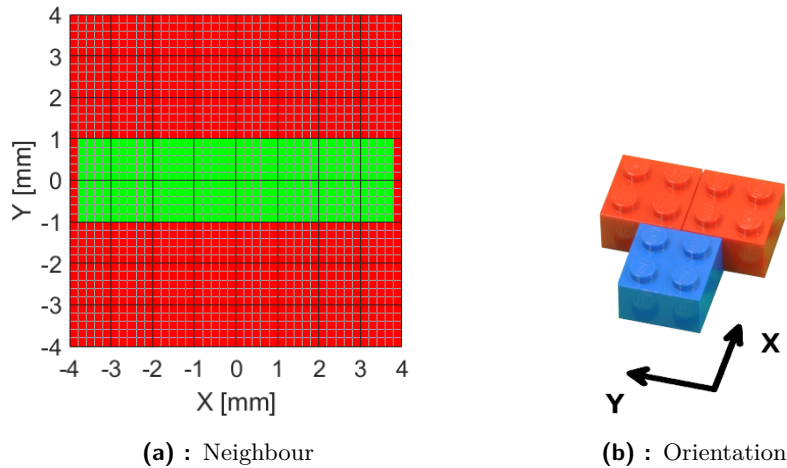


Figure 5.3: Neighbour configuration

The figure (5.4) shows the Corridor configuration for the double Spiral movement, which performs two spiral motions consecutively.

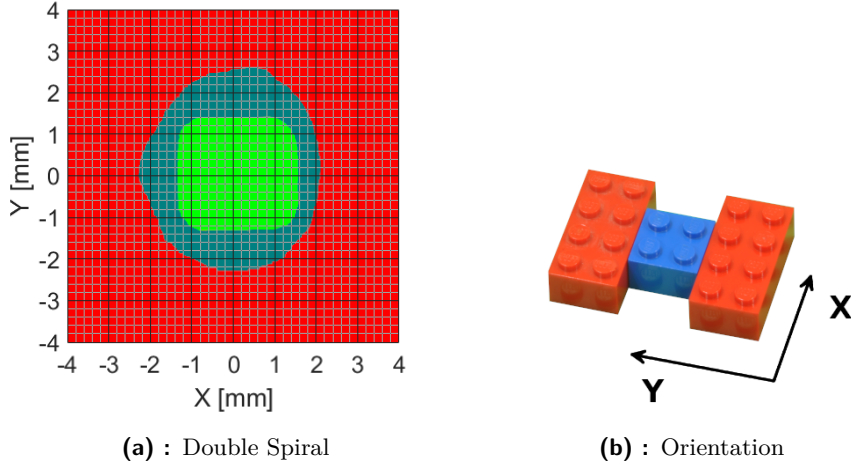


Figure 5.4: Corridor configuration

The figure (5.5) shows the Corner configuration. There are two images. There is an absence of an orientation image because the movement images are both point symmetrical due to the *Touching* motions. It is shown that the designed movement for the Corner configuration has better results than the other. However, both movements are highly successful.

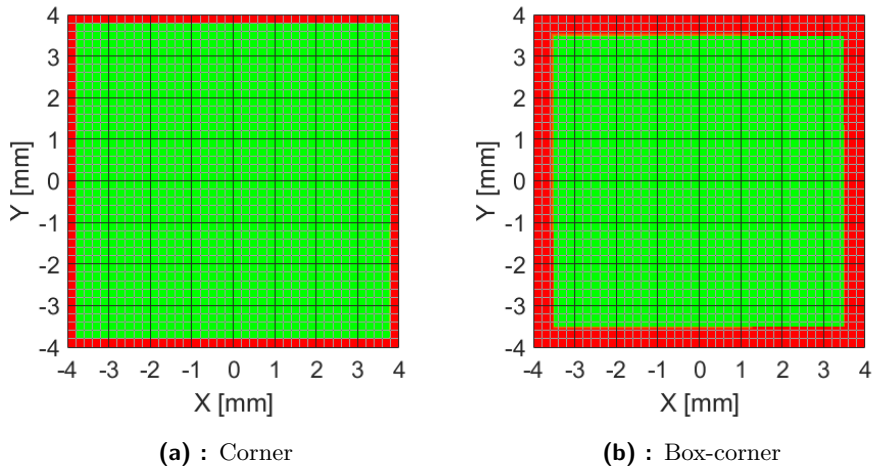


Figure 5.5: Corner configuration

The figure (5.6) shows the Horseshoe configuration and its designed movement. In image (a), there are niches, where the *Spiral* motion of the Horseshoe movement has problems with the corners created by the stud-walls of the

adjacent bricks.

The deviation in the X-negative direction can be almost arbitrary, thanks to the *Touching* motion.

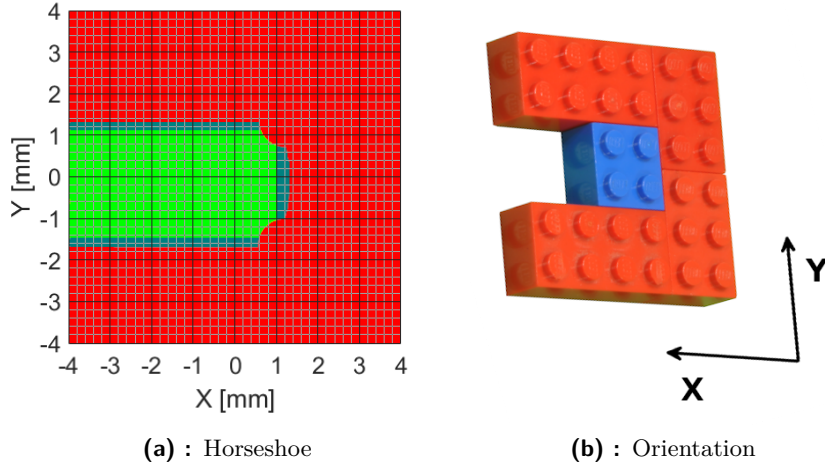


Figure 5.6: Horseshoe configuration

The figure (5.7) shows the Box configuration. The example-1 has no distinguishable orientation. That is the reason why an orientation image is not present here. Both images of the figure depict the designed movements for this configuration. It shows that the Box-corner movement is more successful than the Box-spiral movement.

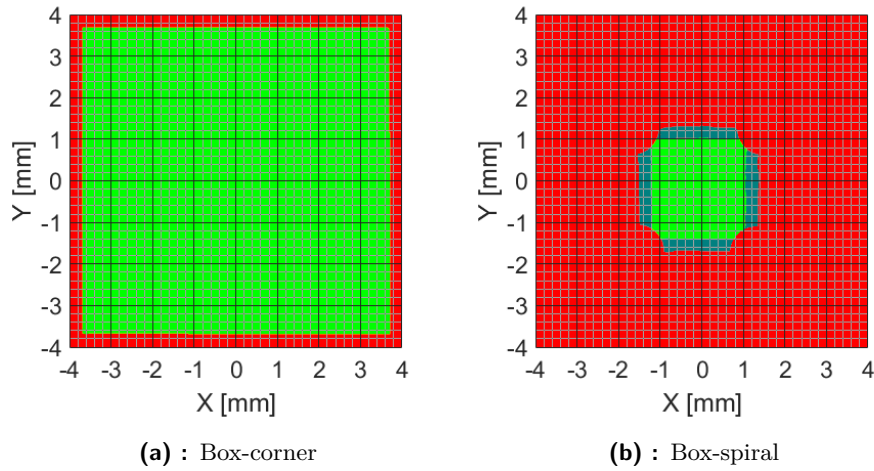


Figure 5.7: Box configuration

Disassembling Experiments

The figure (5.8) shows the Standalone configuration. The used brickwork was made of three 8x2 bricks creating a column to avoid multiple detaching. In the figure, there are three movements. It is shown that the Tilt-Repairing movement is successful in positions where the Tilt movement is at least partially successful.

In the case of the Column configuration (4.3.2), the results of the Tilt and the Tilt-Repairing movements would look similar. The result of the Trivial movement would be dark green instead of green, due to multiple bricks detaching.

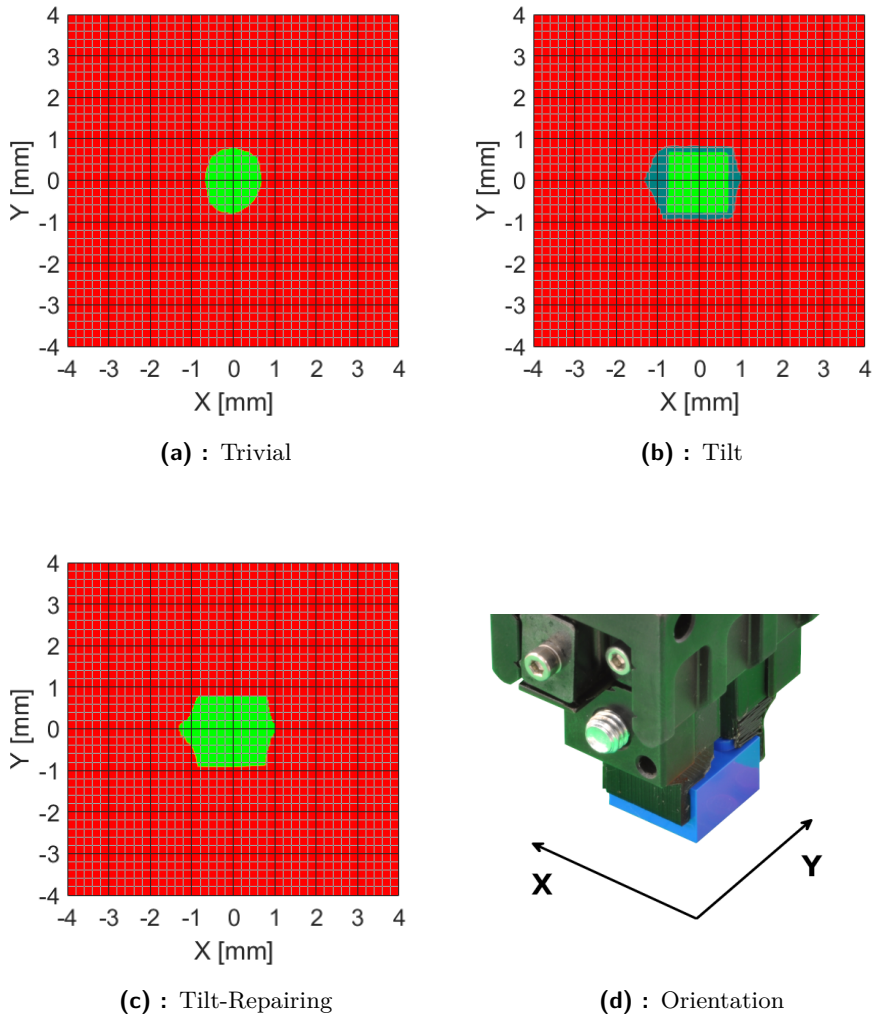


Figure 5.8: Standalone configuration

The figure (5.9) shows the Staircase configuration. The Trivial motion almost always detaches multiple bricks, while the Tilt movement is too strong in the *leveling* motion and partially detaches the brickwork. That is the reason for the dark green labelling.

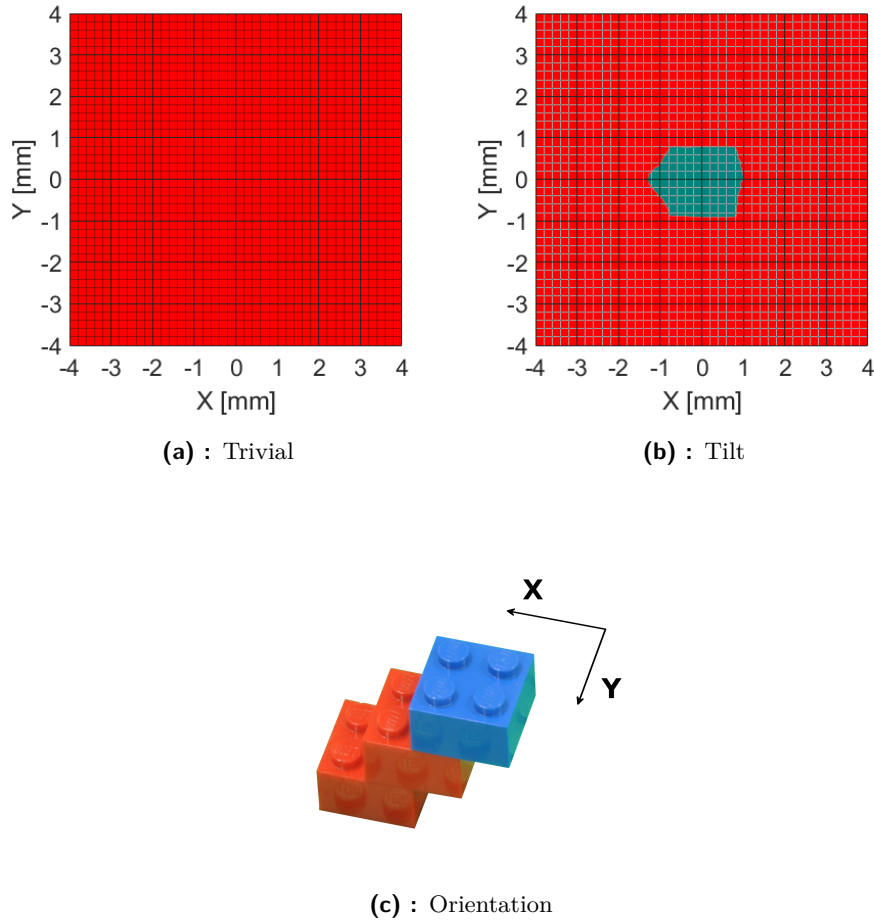


Figure 5.9: Staircase configuration

The figure (5.10) shows the Neighbour configuration and the Tilt movement. It shows that the deviation space is similar to the Standalone configuration but space is limited by the adjacent bricks.

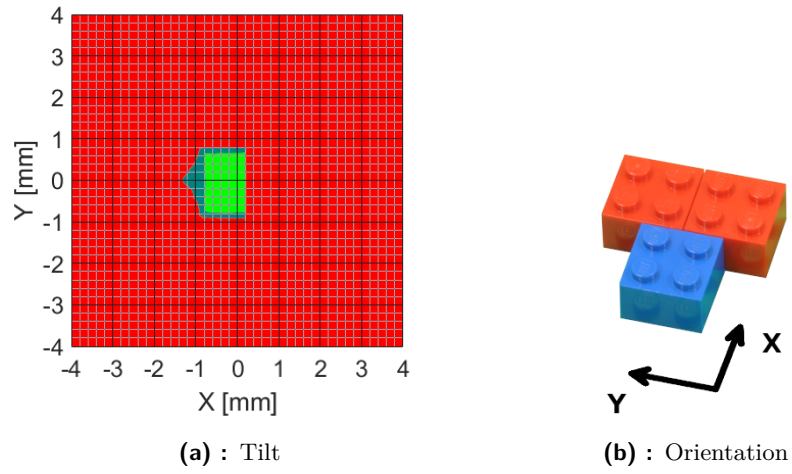


Figure 5.10: Neighbour configuration

It was experimentally tested that the Tilt motions with an equal setting can handle a LEGO brick up to 8x2 size. The attaching Z-downwards force needs to be increased in order to perform this.



Chapter 6

Conclusion

This thesis proposes movements for the assembling and disassembling of LEGO bricks by a compliant robot. The bricks have the size of 2x2 and they are manipulated by fingers which have been specially designed for this task. The issue of the task is in the precise LEGO brick manipulation by an imprecise robot. The imprecision in the position is compensated for the force feedback. This technique can successfully handle relatively high position deviations. Part of the thesis is the proposed terminology of LEGO bricks and the sorting of brick layouts into the placing configurations. The experimental results show that the designed movements for those configurations where the goal position is surrounded by adjacent bricks can overcome higher position deviations than the movements designed for open space configurations.



Bibliography

- [1] J  r  my Brouillard *Assembly of LEGO bricks using a FANUC M-1iA Delta robot*. YouTube, 2015, <https://www.youtube.com/watch?v=BNP74352vhg>
- [2] H. Bruyninckx, S. Dutr   and J. De Schutter *Peg-on-Hole: A Model Based Solution to Peg and Hole Alignment*. IEEE International Conference on Robotics and Automation, 1995.
- [3] Hyeonjun Park, Jaeheung Park, Dong-Hyuk Lee, Jae-Han Park, Moon-Hong Baeg, Ji-Hun Bae *Compliance-Based Robotic Peg-in-Hole Assembly Strategy Without Force Feedback*. IEEE transactions on industrial electronics, vol.64, no. 8, August 2017.
- [4] Hyeonjun Park, Jaeheung Park, Dong-Hyuk Lee, Jae-Han Park, Moon-Hong Baeg, Ji-Hun Bae *Intuitive Peg-in-Hole Assembly Strategy with a Compliant Manipulator* . IEEE ISR 2013, 2014, ISBN: 978-1-4799-1173-8
- [5] Zhi Liu, Libin Song, Zhimin Hou, Ken Chen, Shaoli Liu, Jing Xu *Screw Insertion Method in Peg-in-Hole Assembly for Axial Friction Reduction*. IEEE Access (Volume:7), 2019. ISSN: 2169-3536
- [6] L.L. Lin, Y. Yang, Y.T. Song, B. Nemec, A. Ude, J.A. Rytz, A.G. Buch, N. Kr  ger, T.R. Savarimuthu *Peg-in-Hole assembly under uncertain pose estimation*. Proceeding of the 11th World Congress on Intelligent Control and Automation, 2014. ISBN: 978-1-4799-5825-2
- [7] Z.-W. Luo, K. Ito, M. Ito *Multiple robot manipulators' cooperative compliant manipulation on dynamical environments*. Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '93), 1993. ISBN: 0-7803-0823-9
- [8] Damith Suresh Chathuranga, Zhongkui Wang, Yohan Noh, Thrishantha Nanayakkara, Shinichi Hirai. *A Soft Three Axis Force Sensor Useful for Robot Grippers*. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016. ISSN: 2153-0866
- [9] Y. Aiyama, T. Arai. *Graspless Manipulation with Sensor Feedback*. Proceedings of the 1997 IEEE International Symposium on Assembly and

Task Planning (ISATP'97) - Towards Flexible and Agile Assembly and Manufacturing. ISBN: 0-7803-3820-0.

- [10] LEGO System A/S *Classic Brick Separator*. *The LEGO Group*, 2019. <https://www.lego.com/en-us/roduct/brick-separator-630>
- [11] Christoph Bartneck *LEGO Brick Dimensions & Measurement. Elegant Themes*, 21.4.2019. <http://www.bartneck.de/wp-content/uploads/2019/04/lego-2x4-brick-dimensions-measurements-3001.pdf>
- [12] Nick Turo-Shields *Lego Brick 2x2. GrabCAD, a STRATASYS solution*, 5.6.2016. <https://grabcad.com/library/lego-brick-2x2-2>
- [13] LEGO System A/S *Classic Green Baseplate*. *The LEGO Group*, 2019. <https://www.lego.com/en-us/product/green-baseplate-10700>
- [14] SCHUNK GmbH & Co. KG *EGP 50-N-N-B. SCHUNK GmbH & Co. KG*, 2019. https://www.schunk.com/cz_en/gripping-systems/product/39759-0310960-egp-50-n-n-b
- [15] KUKA Roboter GmbH *KUKA Sunrise.OS 1.14. KUKA Roboter GmbH*, p.508, 14.8.2017.